

# A New Family of Unitary Space-Time Codes with a Fast Parallel Sphere Decoder Algorithm \*

Xinjia Chen, Kemin Zhou and Jorge Aravena

June 2007

## Abstract

In this paper we propose a new design criterion and a new class of unitary signal constellations for differential space-time modulation for multiple-antenna systems over Rayleigh flat-fading channels with unknown fading coefficients. Extensive simulations show that the new codes have significantly better performance than existing codes. We have compared the performance of our codes with differential detection schemes using orthogonal design, Cayley differential codes, fixed-point-free group codes and product of groups and for the same bit error rate, our codes allow smaller signal to noise ratio by as much as 10 dB.

The design of the new codes is accomplished in a systematic way through the optimization of a performance index that closely describes the bit error rate as a function of the signal to noise ratio. The new performance index is computationally simple and we have derived analytical expressions for its gradient with respect to constellation parameters.

Decoding of the proposed constellations is reduced to a set of one-dimensional closest point problems that we solve using parallel sphere decoder algorithms. This decoding strategy can also improve efficiency of existing codes.

## 1 Introduction

Recently there have been extensive research interests in wireless communication links with multiple transmitter antennas. For the Rayleigh-fading

---

\*The authors are with Department of Electrical and Computer Engineering, Louisiana State University, Baton Rouge, LA 70803; Email: {chan, kemin, aravena}@ece.lsu.edu, Tel: (225)578-{8961, 5533,5537}, and Fax: (225) 578-5200.

channel models, information-theoretic analysis has shown that the capacity of a communication link with multiple transmitter antennas can substantially exceed that of a single-antenna link [10], [11], [31], [46], [55]. Several coding and modulation schemes have also been proposed to exploit the potential increase in the capacity through space diversity. For the coherent multiple-antenna channel, several transmit diversity methods and code construction have been presented in [3], [42], [43] and the references therein (see, e.g., [7], [12], [14], [17], [34]–[36], [38], [47], [50]–[53]). In particular, Tarokh, Seshadri, and Calderbank [42] proposed space-time codes which combine signal processing at the receiver with coding techniques appropriate to multiple transmitter antennas. Alamouti [3] discovered a remarkable transmitter diversity scheme for two transmitter antennas, which was later generalized by Tarokh *et al.* [43] as a framework for space-time block codes. Motivated by the fact that, in many situations, channel state information may not be available to the receiver, Hochwald and Marzetta [21] proposed a general signaling scheme, called unitary space-time modulation, and showed that this scheme can achieve a high ratio of channel capacity in combination with channel coding. The design of unitary space-time constellations was investigated in [1], [18] and [19]. More recently, differential modulation and code construction methods for multiple transmit antennas have been proposed by Hochwald *et al.* [20], Hughes [22], Tarokh *et al.* [25, 44] and some other researchers (see, e.g., [4], [16], [23], [24], [27], [30], [32], [40, 41], [49], [54]).

We investigate the encoding and decoding issues for the differential unitary space-time modulation scheme independently proposed by Hochwald and Sweldens in [20] and Hughes in [22]. A number of unitary space-time codes have been proposed aimed at achieving high performance, low encoding and decoding complexity. Among these, we recall the orthogonal design (see, [25, 44]), cyclic group codes [20, 22], Caley differential (CD) codes [16] and the *full-diversity* codes such as fixed-point-free (FPF) unitary group codes  $G_{m,r}$ , non-group codes  $S_{m,s}$  and products of cyclic groups [39]. Orthogonal design has extremely low decoding complexity; unfortunately, the performance degrades significantly when the number of receiver antennas is more than one or the data rate is high. Caley differential codes and the full-diversity codes outperform orthogonal designs in many cases, while the decoding complexity is much higher than that of orthogonal designs. The main idea of decoding the full-diversity codes and Caley differential codes is to formulate the decoding problem as a closest point problem and then solve it by existing methods such as “LLL” lattice algorithm and sphere decoder algorithm. The decoding complexity depends critically on the dimension of

the underlying closest point problem.

In this paper we develop a new paradigm for the design of high performance, low encoding and decoding complexity, unitary space-time codes. Similar to the full-diversity codes  $G_{m,r}$ ,  $S_{m,s}$  and products of cyclic groups, our proposed constellations also use diagonal matrices as the kernel for fast decoding purpose. However, in sharp contrast to those existing codes which are parameterized by special integers, our constellations are defined by real-valued parameters and are not restricted to have *full diversity* or group structure. Consequently, unitary space-time code with our proposed structure exists for any combination of antennas and constellation size.

We define a code performance index that describes the bit error rate as a function of the signal to noise ratio. The index is simple to evaluate yet highly accurate in the normal signal to noise ratio (SNR) region. As a result, it is possible to bring all the power of non-linear programming into the code design. We have developed a complete gradient descent algorithm to design constellations that are optimal with respect to the bit error rate. It should be noted that the idea of code design by gradient-based optimization for non-coherent MIMO channels was proposed before in [1] and [16]. A systematic design of unitary constellation based on random search has been proposed in [19]. Our approach differs from the previous works in the design criterion, the structure of signal constellations, and the decoding method. We attempt to apply gradient descent techniques to directly minimize the bit error rate over signal constellations which allow for efficient decoding algorithms.

Exploiting the special structure of our proposed constellations, the decoding problem is reduced to one-dimensional closest point problems which can be efficiently solved in parallel. Based on that strategy, we have developed parallel sphere decoder algorithms which can also be applied to improve decoding efficiency of existing codes.

Based on the new structure and using the optimal design techniques, we have obtained constellations which significantly outperform existing ones. For example, with spectral efficiency  $R = 6$  bits per channel use, we have found a constellation which improves upon orthogonal design by about 10 dB at block error rate  $6 \times 10^{-2}$  when using two transmitter and receiver antennas. With the same configuration, the corresponding improvement upon Caley differential code is about 9 dB.

In the rest of this section we establish the notation and describe the channel model. Section 2 introduces the structure of the new constellations and develops the optimization procedure for their design. Specifically, we introduce the performance index that converts constellation design into a

minimization problem amenable to steepest descent techniques and derive simple expressions for the computation of its gradient. Section 3 develops a parallel sphere decoder algorithm that can also be applied to improve existing codes. Section 4 presents results of the performed simulations. Section 5 summarizes our findings. Proofs and constellation data are provided in the Appendices.

## 1.1 Notation

Throughout this paper, we use the following notations.

- $\mathbb{R}$  — real number field;
- $\mathbb{C}$  — complex number field;
- $\mathbb{Z}$  — integer set;
- $\lfloor \cdot \rfloor$  — floor function;
- $\lceil \cdot \rceil$  — ceiling function;
- $\lfloor x \rfloor$  — the integer closest to  $x$ ;
- $\text{mod}^*(x)$  — symmetric modulus operation such that  $\text{mod}^*(x)$  has range  $[-\frac{x}{2}, \frac{x}{2})$ ;
- $\arg(\cdot)$  — phase angle operator taking values in  $[-\pi, \pi)$ ;
- $\det(\cdot)$  — determinant function;
- $\text{tr}(\cdot)$  — trace function;
- $\text{diag}([x_1, \dots, x_n])$  — diagonal matrix with  $x_p$  at the  $p$ -th row and the  $p$ -th column;
- $\|X\|$  — Euclidean norm of vector  $X$ ;
- $\|X\|_F$  — Frobenius norm of matrix  $X$ ;
- $[X]_{pq}$  — entry of  $X$  at the  $p$ -th row and  $q$ -th column;
- $\Re(X)$  — real part of  $X$ ;
- $\Im(X)$  — imaginary part of  $X$ ;
- $X^\top$  — transpose of  $X$ ;
- $X^\dagger$  — conjugate transpose of  $X$ ;
- $\text{abs}(X)$  — the matrix obtained by replacing each entry of  $X$  with its modulus;
- $\mathcal{CN}(0, 1)$  — complex random variable with zero mean and variance one.
- $\nabla g(\mathbf{x})$  — gradient of function  $g(\mathbf{x})$ .

## 1.2 Channel Model

Consider a communication link with  $M$  transmitter and  $N$  receiver antennas operating in a Rayleigh flat-fading channel, which can be described by the

following channel model [20]

$$X_\tau = \sqrt{\rho} S_\tau H_\tau + W_\tau$$

where  $\tau$  is the index of time frame,  $H_\tau \in \mathbb{C}^{M \times N}$  is the channel matrix with  $\mathcal{CN}(0, 1)$  entries and is unknown to the receiver and the transmitter,  $S_\tau \in \mathbb{C}^{M \times M}$  is the transmitted signal,  $X_\tau \in \mathbb{C}^{M \times N}$  is the received signal,  $W_\tau \in \mathbb{C}^{M \times N}$  is Gaussian noise with  $\mathcal{CN}(0, 1)$  entries, and  $\rho$  is the expected SNR at each receiver antenna. It should be noted that the channel matrix  $H_\tau$  has been normalized so that the SNR is not dependent on the number of transmitter antennas. It is assumed that the channel matrix is approximately constant within two consecutive time frames, i.e.,  $H_\tau \approx H_{\tau-1}$ . However, for the  $\tau$ -th and the  $\iota$ -th time frames that are not consecutive,  $H_\tau$  and  $H_\iota$  are mutually independent and thus their realizations can be significantly different. The transmitted signals are determined by the following fundamental differential transmitter equations [20]

$$S_0 = I_{M \times M}, \quad S_\tau = V_\tau S_{\tau-1}, \quad \tau = 1, 2, \dots$$

where  $V_\tau \in \mathbb{C}^{M \times M}$  is a unitary matrix picked from signal constellation  $\mathcal{V}$ . It is shown in [20, 21, 22] that the maximum-likelihood (ML) detection is to minimize

$$\|X_\tau - V_\ell X_{\tau-1}\|_F^2$$

among all possible  $V_\ell \in \mathcal{V}$ . The Chernoff bound of pair-wise probability of mistaking  $V_\ell$  for  $V_{\ell'}$  or vice versa is given by [21]

$$P(V_\ell, V_{\ell'}) = \frac{1}{2} \prod_{m=1}^M \left[ 1 + \frac{\rho^2 \sigma_m^2}{4(1 + 2\rho)} \right]^{-N} \quad (1)$$

where  $\sigma_m$  is the  $m$ -th singular value of  $V_\ell - V_{\ell'}$ .

## 2 A New Constellation Design Approach

The new code design paradigm that we propose uses diagonal matrices, as in [39], to simplify the decoding process. Our approach is similar to [1] and [16] in the spirit of relaxing the code structures from strict structures such as orthogonal or diagonal structure, parameterizing the codes, and employing the powerful gradient-based optimization to find the best codes. A significant new feature is the ability to formulate the design as a non-linear programming problem that directly minimizes the bit error rate. In

the following, we begin our presentation by introducing new codes that are functions of real-valued variables and do not require full diversity. Then we introduce the cost function and derive expressions for its gradient that are used in a steepest descent design algorithm.

## 2.1 Constellation Structure

In this section, we introduce a new class of unitary space-time codes which can be efficiently encoded and decoded. Similar to the full-diversity codes such as FPF code  $G_{m,r}$ , non-group code  $S_{m,s}$  and products of cyclic groups [39], our proposed constellation also involves diagonal matrices for fast decoding purpose. However, in sharp contrast to those full-diversity codes which are parameterized by particular integers, our proposed constellations are determined by continuous parameters and are not restricted to have full diversity or group structure. Consequently, unitary space-time code with our proposed structure exists for any combination of antennas and constellation size.

Let  $b \geq 0$  be an integer and let  $L$  be a power of 2. We construct a constellation  $\mathcal{V}$  with  $\mathcal{L} = 2^b L$  signal matrices as follows.

For  $q = 0, 1, \dots, 2^b - 1$ , define

$$\Lambda_q \stackrel{\text{def}}{=} \text{diag} \left( \left[ \exp \left( \frac{i2\pi\lambda_{q,1}}{L} \right), \dots, \exp \left( \frac{i2\pi\lambda_{q,M}}{L} \right) \right] \right),$$

where  $\lambda_{q,1} = 1$  and  $\lambda_{q,m} \in [0, L)$ ,  $m = 2, \dots, M$  are *real-valued* parameters. Let  $A_0 = B_0 = I$  and  $A_q, B_q$ ,  $q = 1, \dots, 2^b - 1$  be unitary matrices. Then the constellation is given by

$$\mathcal{V} = \{A_q \Lambda_q^\ell B_q \mid \ell = 0, 1, \dots, L-1; q = 0, 1, \dots, 2^b - 1\}.$$

We note that the constellation design problem is to find  $\lambda_{q,m}$  and  $A_q, B_q$  so that the bit error rate is minimized. We shall show that this problem can be solved efficiently.

For the purpose of comparing our constellations with existing ones, we note that the spectral efficiency of our proposed constellation is

$$R = \frac{\log_2(\mathcal{L})}{M} = \frac{b + \log_2(L)}{M}.$$

It should be noted that, for the special case  $b = 0$ , the signal constellation reduces to

$$\{\Lambda^\ell \mid \ell = 0, 1, \dots, L-1\}$$

where

$$\Lambda = \text{diag} \left( \left[ \exp \left( \frac{i2\pi\lambda_1}{L} \right), \dots, \exp \left( \frac{i2\pi\lambda_M}{L} \right) \right] \right)$$

with  $\lambda_1 = 1$  and *continuous* parameters  $\lambda_m \in [0, L)$ ,  $m = 2, \dots, M$ . We refer to such constellation as a *continuous diagonal code*. Obviously, it is a generalization of cyclic group code.

In general, with fixed constellation size  $\mathcal{L}$ , the performance may be significantly improved by increasing the number of blocks (i.e.,  $\frac{\mathcal{L}}{L}$ ). Interestingly, we shall show that the decoding complexity increases slightly with respect to the number of blocks. This property can be attributed to our parallel sphere decoder algorithms, discussed in Section 3.

## 2.2 Design Performance Index

Efficient constellation design is a challenging task due to the large number of parameters. In addition to the structure of the constellations, the design criterion is also critical for the achievable bit error rate performance. One of the widely used criterion is to use the diversity product as the performance measure of a constellation. The design objective is to maximize the diversity product over a class of constellations that have full diversity (see, e.g., [20] [27], [39] and the references therein). The drawbacks of the conventional design criterion are the following: First, the diversity product is essentially a worst-case measure. In many situations, the overall performance of a constellation is not governed by the behavior of extreme signal matrices. As can be seen from our experimental results in Section 4, it is not uncommon to have constellations with *zero diversity product* significantly outperforming constellations with the largest diversity product previously known. Second, the measure diversity product is derived by an asymptotic argument. The idea is that, as the SNR tends to infinity, the Chernoff bound of the pair-wise error probability is dominated by the determinant of the difference of the pair of unitary matrices. Such asymptotic argument is not flawless. It is not clear how large the value of SNR can be approximated as infinity so that no significant inaccuracy will be introduced in the evaluation of the block (or bit) error rate.

In light of the limitations of the worst-case and asymptotic design criterion, we have established a new design criterion which incorporates the bits assignment in the optimization of constellations. Instead of using a worst-case criterion such as diversity product [20], we introduce a performance index that measures directly the bit error rate as a function of the signal

to noise ratio. The index is analytically tractable and possesses simple analytical expressions for its gradient. Motivated by the fact that, for large constellation size, the bit error rate may not be well governed by the block error rate, we shall also incorporate the bit assignments in the process of constellation optimization. In particular we propose the cost function

$$J = \int_{\rho_1}^{\rho_2} \log_{10} P_{\text{bit}}(\rho) d \log_{10}(\rho)$$

where  $P_{\text{bit}}(\rho)$  is the union bound of bit error probability and  $[\rho_1, \rho_2]$  is the interval of SNR of practical interests. We shall show that this cost function can be well approximated by a very simple analytical expression.

From numerous simulation results published in the literature, we notice that, on a log scale, the bit error rate is an almost linear function of the SNR. Such phenomenon can be illustrated by making use of the Chernoff bound (1). For large SNR, the Chernoff bound  $P(V_\ell, V_{\ell'})$  of pair-wise error probability can be approximated by

$$\begin{aligned} P(V_\ell, V_{\ell'}) &\approx \frac{1}{2} \rho^{-MN} \left( \prod_{m=1}^M \frac{\sigma_m^2}{8} \right)^{-N} \\ &= \frac{1}{2} \rho^{-MN} \left( \frac{\det(V_\ell - V_{\ell'})}{8} \right)^{-N}. \end{aligned}$$

Since such approximation is tight for most combinations  $(\ell, \ell')$  and  $V_\ell$  is assumed to be equally likely for all  $\ell$ , the union bound of the bit error rate is well approximated by

$$P_{\text{bit}}(\rho) \approx \rho^{-MN} \frac{\sum_{\ell \neq \ell'} d^{\text{H}}(\ell, \ell') \left( \frac{\det(V_\ell - V_{\ell'})}{8} \right)^{-N}}{2\mathcal{L} \log_2 \mathcal{L}}$$

where  $d^{\text{H}}(\ell, \ell')$  denotes the Hamming distance of bits assigned to  $V_\ell$  and  $V_{\ell'}$ . Applying logarithm operation gives

$$\begin{aligned} &\log_{10} P_{\text{bit}}(\rho) \\ &\approx -\frac{MN}{10} \log_{10}(\rho) \\ &\quad + \log_{10} \left( \frac{\sum_{\ell \neq \ell'} d^{\text{H}}(\ell, \ell') \left( \frac{\det(V_\ell - V_{\ell'})}{8} \right)^{-N}}{2\mathcal{L} \log_2 \mathcal{L}} \right). \end{aligned}$$



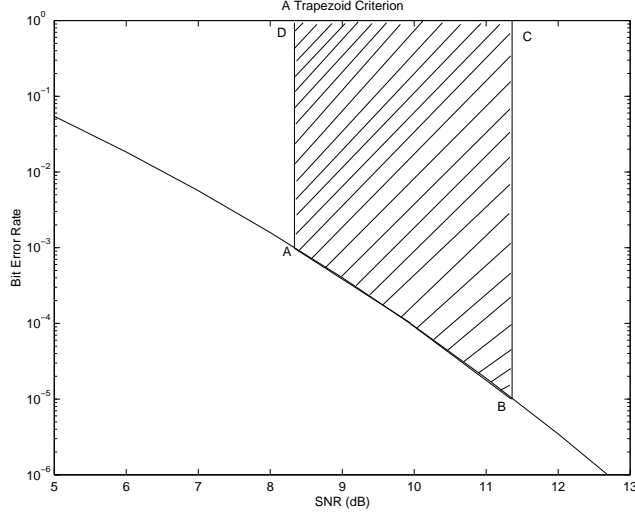


Figure 1: The area of trapezoid ABCD, or equivalently  $-\zeta(\mathcal{V})$ , reflects the quality of constellation  $\mathcal{V}$ .

Figure 1 displays the actual cost function and the proposed approximation. For completeness we mention that the block error rate admits a similar approximation.

Due to the excellent linearity of the performance curve in the logarithm scale, the cost function can be well approximated by

$$\zeta(\mathcal{V}) \stackrel{\text{def}}{=} [\log_{10} P_{\text{bit}}(\rho_2) + \log_{10} P_{\text{bit}}(\rho_1)] \times [\log_{10}(\rho_2) - \log_{10}(\rho_1)].$$

We propose to design constellations that minimize the index  $\zeta(\mathcal{V})$ .

In practice, we can choose  $\rho_1$  and  $\rho_2$  based on the performance of the best cyclic group codes previously known. More specifically,  $\rho_1$  and  $\rho_2$  can be selected so that two typical levels of bit error rate are respectively guaranteed. For example, we can find  $\rho_1$  and  $\rho_2$  such that

$$\log_{10} P_{\text{bit}}(\rho_1) = 10^{-3}, \quad \log_{10} P_{\text{bit}}(\rho_2) = 10^{-5}$$

by a bisection method for an existing cyclic group code. When  $\rho_1$  and  $\rho_2$  have been found, the criterion measure  $\zeta(\mathcal{V})$  is SNR independent. Most importantly, the gradient of  $\zeta(\mathcal{V})$  with respect to the constellation parameters can be computed efficiently and thus allows for a gradient descent method for constellation design. The optimization technique is described in the next section.

## 2.3 Constellation Optimization

In this section, we perform a global optimization to find unitary constellations of good performance. Our strategy is to first choose the bit assignment and then search the code parameters to minimize the bit error rate. The advantage of this strategy is that the objective function  $\zeta(\mathcal{V})$  is a differentiable function and is amenable for gradient-based optimization. On the other hand, if we first search the good code matrices and then try to find the best bit assignment, we need to solve a combinatorial optimization problem. In general, such combinatorial problem is not tractable for gradient-based optimization techniques because the objective function is not continuous. The only method for solving such combinatorial problem is the exhaustive random search. Unfortunately, for large constellations, the searching can be extremely inefficient.

### 2.3.1 Parameterization of Unitary Matrix

In order to develop a gradient-based method for the minimization of the performance measure  $\zeta(\mathcal{V})$ , the first step is to choose a suitable parameterization for unitary matrices. The application of unitary matrices parameterization [33] in signal constellation design has been pioneered by [1]. We adopt such idea of using parameterized unitary code matrices. In general, a  $M \times M$  unitary matrix  $U$  can be determined by a set of  $M^2$  parameters  $\Theta$  defined as follows:

$$\begin{aligned}\phi_{pq} &\in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right], & 1 \leq p < q \leq M-1; \\ \phi_{pM} &\in [-\pi, \pi), & 1 \leq p \leq M-1; \\ \nu_{pq} &\in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right], & 1 \leq p < q \leq M; \\ \theta_k &\in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right], & k = 1, \dots, M-1; \\ \theta_M &\in [-\pi, \pi).\end{aligned}$$

More specifically, let  $U^{p,q}(\phi_{pq}, \nu_{pq})$  denote a  $(M-p+1)$ -dimensional unitary matrix such that

$$[U^{p,q}]_{jk} = \begin{cases} 1, & \text{if } j = k \text{ and } j \notin \{1, q-p+1\} \\ \cos(\phi_{pq}), & \text{if } j = k \text{ and } j \in \{1, q-p+1\} \\ -\sin(\phi_{pq})e^{-i\nu_{pq}}, & \text{if } j = 1 \text{ and } k = q-p+1 \\ \sin(\phi_{pq})e^{i\nu_{pq}}, & \text{if } k = 1 \text{ and } j = q-p+1 \\ 0, & \text{otherwise} \end{cases}$$

and let

$$F^r = U^{r,r+1} U^{r,r+2} \dots U^{r,M},$$

then, any unitary matrix  $U(\Theta)$  can be represented as

$$U = \mathcal{U}^{M-1} F^1$$

where

$$\mathcal{U}^1 = \begin{bmatrix} \exp(i\theta_{M-1}) & 0 \\ 0 & \exp(i\theta_M) \end{bmatrix}$$

and

$$\mathcal{U}^{k+1} = \begin{bmatrix} \exp(i\theta_{M-k-1}) & 0 \\ 0 & \mathcal{U}^k F^{M-k} \end{bmatrix}$$

for  $k = 1, \dots, M-2$ .

### 2.3.2 Gradient Method

Here we develop explicit expressions for the gradient of the performance measure  $\zeta(\mathcal{V})$ . For the computation of  $\zeta(\mathcal{V})$  we need to evaluate the union bound of the bit error rate, which depends on the bit assignment. With regard to the bit assignment, our intuition is that, if we first search the good code matrices and then try to find the best bit pattern – code matrix assignment, we need to cope with a combinatorial optimization problem. Such combinatorial problem is generally not tractable for gradient-based optimization techniques because the objective function is not differentiable. The available method for solving such combinatorial problem will be random search. Unfortunately, for large constellations, the searching can be extremely difficult. In our design, we shall first fix the bit assignment and then search the code parameters to minimize the bit error rate. In this way, the objective function is a differentiable function and is amenable for gradient-based optimization techniques.

For simplicity, we use the binary-to-decimal conversion mapping scheme. In such a scheme, a block of  $b + \log_2(L)$  bits is mapped into a signal matrix  $A_q \Lambda_q^\ell B_q$  such that the first  $b$  bits are the binary representation of the block index  $q$  and the remaining bits are the binary representation of the diagonal index  $\ell$ . Let  $d^H(p, q, \ell, \ell')$  denote the Hamming distance between the bits respectively assigned to signal matrices  $A_p \Lambda_p^\ell B_p$  and  $A_q \Lambda_q^{\ell'} B_q$ . The union bound of the bit error probability is then given by

$$P_{\text{bit}} = \frac{2}{2^b L [b + \log_2(L)]} (\hat{\mathcal{P}} + \tilde{\mathcal{P}}) \quad (2)$$

where

$$\widehat{\mathcal{P}} = \sum_{p=0}^{2^b-1} \sum_{\ell=0}^{L-2} \sum_{\ell'=\ell+1}^{L-1} d^H(p, p, \ell, \ell') P(\Lambda_p^\ell, \Lambda_p^{\ell'})$$

and

$$\begin{aligned} & \widetilde{\mathcal{P}} \\ = & \sum_{p=0}^{2^b-2} \sum_{q=p+1}^{2^b-1} \sum_{\ell=0}^{L-1} \sum_{\ell'=0}^{L-1} d^H(p, q, \ell, \ell') P(A_p \Lambda_p^\ell B_p, A_q \Lambda_q^{\ell'} B_q) \end{aligned}$$

It can be seen that, using (2) to compute  $P_{\text{bit}}$ , the number of pair-wise error probabilities to be evaluated is

$$2^{b-1}L(L-1) + 2^{b-1}(2^b-1)L^2.$$

The problem can still be solved using steepest descent method for small to moderate  $L$ . However, the computational complexity may be high for large  $L$ . For proof of concept, we focus here on the special case of  $\Lambda_q = \Lambda$ ,  $A_q = I$  for  $q = 0, \dots, 2^b-1$ , where, exploiting the special structure of the constellation, the number of pair-wise error probabilities to be computed can be substantially reduced to

$$(L-1) + 2^{b-1}(2^b-1)(2L-1).$$

For this case, we have

**Theorem 1** *Let  $d^H(p, q)$  denote the Hamming distance between the binary representation of integers  $p$  and  $q$ . Define*

$$w(k) = \sum_{\ell=0}^{L-k-1} d^H(\ell+k, \ell), \quad k = 0, 1, \dots, L-1.$$

*Then*

$$P_{\text{bit}} = \frac{2}{L[b + \log_2(L)]} \left[ \mathcal{P}' + \sum_{k=1}^{L-1} w(k) P(I, \Lambda^k) \right]$$

*where*

$$\mathcal{P}' = \frac{1}{2^b} \sum_{p=0}^{2^b-2} \sum_{q=p+1}^{2^b-1} \sum_{k=-L+1}^{L-1} [w(|k|) + d^H(p, q)] P(B_p, \Lambda^k B_q).$$

See Appendix A for a proof. It should be noted that, to reduce computation,  $w(k)$  can be pre-computed and saved as a lookup table.

In order to use gradient descent method to minimize  $\zeta(\mathcal{V})$ , we need to find the fastest descent direction at every step of searching. Following the procedure in [1], we update  $B_p$  as  $B_p U(\Theta)$ . In the sequel, we shall show that the computation of the gradient of performance measure  $\zeta(\mathcal{V})$  reduces to the computation of: (i) the partial derivatives of functions of the form  $P(U(\Theta), \Phi)$  with respect to  $\Theta$  at  $\Theta = 0$  (i.e., all elements of  $\Theta$  are zero); (ii) the partial derivatives of functions of the form  $P(\Lambda^\ell, \Phi)$  with respect to  $\Lambda = \text{diag}([e^{2\pi i \lambda_1/L}, \dots, e^{2\pi i \lambda_M/L}])$  at  $\Lambda = I$  (i.e.,  $\lambda_m = 0$ ,  $m = 1, \dots, M$ ).

We have derived surprisingly simple formulas for computing pair-wise error probabilities and the related partial derivatives.

**Theorem 2** *Let  $U(\Theta)$  be unitary matrix parameterized by  $\Theta$ . Let  $\Phi$  be a unitary matrix. Let*

$$\alpha = \frac{4(1+2\rho)}{\rho^2}, \quad \mathcal{Q} = [(\alpha+2)I - \Phi - \Phi^\dagger]^{-1}\Phi$$

and

$$\Lambda = \text{diag}([e^{2\pi i \lambda_1/L}, \dots, e^{2\pi i \lambda_M/L}]).$$

Then

$$P(I, \Phi) = \frac{\alpha^{MN}}{2(\det[(\alpha+2)I - \Phi - \Phi^\dagger])^N}, \quad (3)$$

$$\left. \frac{\partial P(U, \Phi)}{\partial \phi_{pq}} \right|_{\Theta=0} = 2NP(I, \Phi) \Re([\mathcal{Q}]_{qp} - [\mathcal{Q}]_{pq}), \quad (4)$$

$$\left. \frac{\partial P(U, \Phi)}{\partial \theta_k} \right|_{\Theta=0} = 2NP(I, \Phi) \Im([\mathcal{Q}]_{kk}), \quad (5)$$

$$\left. \frac{\partial P(U, \Phi)}{\partial \nu_{pq}} \right|_{\Theta=0} = 0, \quad (6)$$

$$\left. \frac{\partial P(\Lambda^\ell, \Phi)}{\partial \lambda_m} \right|_{\Lambda=I} = \frac{4\pi N \ell}{L} P(I, \Phi) \Im([\mathcal{Q}]_{mm}). \quad (7)$$

See Appendix B for a proof. At the first glance, it is not clear how Theorem 2 can be applied to the optimization of code matrices. From the expression of our performance metric  $\zeta(\mathcal{V})$ , it can be seen that it suffices

to compute the gradients of  $P_{\text{bit}}(\rho_1)$  and  $P_{\text{bit}}(\rho_2)$  with respect to code parameters. From (2), we can see that, since the bits assignment is fixed, it suffices to compute the gradient of  $P(\Lambda_p^\ell, \Lambda_p^{\ell'})$  and  $P(A_p \Lambda_p^\ell B_p, A_q \Lambda_q^{\ell'} B_q)$  with respect to code parameters for all combinations of  $p$  and  $q$ . Note that the first quantity can be viewed as a special case of the second. Hence, we focus on the second quantity  $P(A_p \Lambda_p^\ell B_p, A_q \Lambda_q^{\ell'} B_q)$ . We first consider how to update the matrix  $B_p$ . In order to apply the gradient descent method to minimize the performance metric, we need the partial derivatives of the function  $P(A_p \Lambda_p^\ell B_p, A_q \Lambda_q^{\ell'} B_q)$  with respect to the parameters of  $B_p$ . It can be seen from the complexity of the function  $P(.,.)$  and the parameterization of  $B_p$  that the direct computation of the partial derivatives can be extremely difficult. Observing that, at every step,  $B_p$  is to be updated as  $\hat{B}_p$  which is also a unitary matrix. Hence, there must be an unitary matrix  $U(\Theta)$  such that  $\hat{B}_p = B_p U(\Theta)$ . This means that we can update the unitary matrices in a multiplicative way. As mentioned earlier, this method of updating unitary matrices was proposed in [1]. In the same spirit with that of the conventional steepest-descent minimization, to make  $P(A_p \Lambda_p^\ell B_p, A_q \Lambda_q^{\ell'} B_q)$  descent in a fastest way as  $B_p$  is varying to a new matrix, we can choose  $U(\Theta)$  based on the partial derivatives of  $P(A_p \Lambda_p^\ell B_p U(\Theta), A_q \Lambda_q^{\ell'} B_q)$  with respect to  $\Theta$  at  $\Theta = 0$ . The computation of the derivatives can be accomplished by applying Theorem 2 and the following fact:

*$P(.,.)$  is invariant under unitary transforms. That is, for any unitary matrices  $X$  and  $Y$ ,  $P(U_L X U_R, Y) = P(X, U_L^\dagger Y U_R^\dagger)$  for any unitary matrices  $U_L$  and  $U_R$ .*

To prove this fact, we can use equation (28), which is shown in Appendix B. By (28),

$$\begin{aligned} & P(U_L X U_R, Y) \\ &= \frac{\alpha^{MN}}{2 (\det[\alpha I + (U_L X U_R - Y)(U_L X U_R - Y)^\dagger])^N}. \end{aligned}$$

Observing that

$$\begin{aligned} & (U_L X U_R - Y)(U_L X U_R - Y)^\dagger \\ &= U_L (X - U_L^\dagger Y U_R^\dagger) U_R U_R^\dagger (X - U_L^\dagger Y U_R^\dagger)^\dagger U_L^\dagger \\ &= U_L (X - U_L^\dagger Y U_R^\dagger) (X - U_L^\dagger Y U_R^\dagger)^\dagger U_L^\dagger, \end{aligned}$$

we have

$$\alpha I + (U_L X U_R - Y)(U_L X U_R - Y)^\dagger$$

$$= U_L \left[ \alpha I + (X - U_L^\dagger Y U_R^\dagger)(X - U_L^\dagger Y U_R^\dagger)^\dagger \right] U_L^\dagger.$$

Hence

$$\begin{aligned} & \det \left[ \alpha I + (U_L X U_R - Y)(U_L X U_R - Y)^\dagger \right] \\ = & \det(U_L U_L^\dagger) \det \left[ \alpha I + (X - U_L^\dagger Y U_R^\dagger)(X - U_L^\dagger Y U_R^\dagger)^\dagger \right] \\ = & \det \left[ \alpha I + (X - U_L^\dagger Y U_R^\dagger)(X - U_L^\dagger Y U_R^\dagger)^\dagger \right] \end{aligned}$$

and

$$\begin{aligned} & P(U_L X U_R, Y) \\ = & \frac{\alpha^{MN}}{2 \left\{ \det \left[ \alpha I + (X - U_L^\dagger Y U_R^\dagger)(X - U_L^\dagger Y U_R^\dagger)^\dagger \right] \right\}^N} \\ = & P(X, U_L^\dagger Y U_R^\dagger). \end{aligned}$$

This proves the invariant property. An immediate result from such property is

$$P \left( A_p \Lambda_p^\ell B_p, A_q \Lambda_q^{\ell'} B_q \right) = P \left( I, (A_p \Lambda_p^\ell B_p)^\dagger A_q \Lambda_q^{\ell'} B_q \right),$$

which implies that we can let  $\Phi = (A_p \Lambda_p^\ell B_p)^\dagger A_q \Lambda_q^{\ell'} B_q$  and apply (3) to compute  $P(A_p \Lambda_p^\ell B_p, A_q \Lambda_q^{\ell'} B_q)$ .

Making use of such property, we have

$$\begin{aligned} & P \left( A_p \Lambda_p^\ell B_p U(\Theta), A_q \Lambda_q^{\ell'} B_q \right) \\ = & P \left( U(\Theta), (A_p \Lambda_p^\ell B_p)^\dagger A_q \Lambda_q^{\ell'} B_q \right). \end{aligned}$$

If we identify  $(A_p \Lambda_p^\ell B_p)^\dagger A_q \Lambda_q^{\ell'} B_q$  as  $\Phi$ , we have

$$P \left( A_p \Lambda_p^\ell B_p U(\Theta), A_q \Lambda_q^{\ell'} B_q \right) = P(U(\Theta), \Phi).$$

Hence, the partial derivatives of  $P(A_p \Lambda_p^\ell B_p U(\Theta), A_q \Lambda_q^{\ell'} B_q)$  can be computed by applying Theorem 2.

Similarly, we can update  $A_p$  as  $A_p U(\Theta)$  and compute the partial derivatives of

$$\begin{aligned} & P \left( A_p U(\Theta) \Lambda_p^\ell B_p, A_q \Lambda_q^{\ell'} B_q \right) \\ = & P \left( U(\Theta), A_p^\dagger A_q \Lambda_q^{\ell'} B_q (\Lambda_p^\ell B_p)^\dagger \right) \end{aligned}$$

with respect to  $\Theta$  at  $\Theta = 0$ . The calculation can be done by identifying  $A_p^\dagger A_q \Lambda_q^{\ell'} B_q (\Lambda_p^\ell B_p)^\dagger$  as  $\Phi$  and applying Theorem 2.

In the same spirit, we can update  $\Lambda_p^\ell$  as  $(\Lambda_p \Lambda)^\ell$  and compute the partial derivatives of

$$\begin{aligned} & P \left( A_p (\Lambda_p \Lambda)^\ell B_p, A_q \Lambda_q^{\ell'} B_q \right) \\ &= P \left( \Lambda^\ell, (A_p \Lambda_p^\ell)^\dagger A_q \Lambda_q^{\ell'} B_q B_p^\dagger \right) \end{aligned}$$

with respect to  $\Lambda$  at  $\Lambda = I$  (i.e.,  $\lambda_m = 0, m = 1, \dots, M$ ). This can be accomplished by letting

$$\Phi = (A_p \Lambda_p^\ell)^\dagger A_q \Lambda_q^{\ell'} B_q B_p^\dagger$$

and invoking Theorem 2.

Finally, because of symmetry, we have

$$P \left( A_p \Lambda_p^\ell B_p, A_q \Lambda_q^{\ell'} B_q \right) = P \left( A_q \Lambda_q^{\ell'} B_q, A_p \Lambda_p^\ell B_p \right).$$

Hence, we can update matrices  $A_q$ ,  $\Lambda_q$  and  $B_q$  and compute the corresponding partial derivatives by the similar method as that of matrices  $A_p$ ,  $\Lambda_p$  and  $B_p$ .

In the gradient-based optimization, we used the standard steepest gradient descent method in [56], with some minor modification to adapt to parameter bounds. In the course of experimenting with the new design paradigm, we have observed that it is beneficial to apply the following searching strategy.

STEP (a). Find the best constellation of diagonal structure  $\{\Lambda^\ell \mid 0 \leq \ell \leq L-1\}$ . This can be done as follows. First, perform random search to find  $n$  good initial values of  $\Lambda$ . Second, for each initial value of  $\Lambda$ , perform gradient-based optimization. Finally, choose the best one among the  $n$  outcomes.

STEP (b). Let  $\Lambda$  be found in the first step. Find the best constellation of special structure  $\{\Lambda^\ell B_q \mid \ell = 0, \dots, L-1; q = 0, \dots, 2^b-1\}$  by employing gradient descent search over  $B_q$  while  $\Lambda$  is fixed. Here the initial value of  $B_q$  can be randomly chosen.

STEP (c). Using the code found at the second step as starting point, search  $A_q$ ,  $B_q$  and  $\Lambda_q$  by gradient descent method. Here the initial value of  $A_q$  can be randomly chosen.



For Steps (a)-(c) in the above strategy, we have adopted the same choice of the step size as that of the algorithm of [56].

### 3 Fast Decoding

Now that we have efficient constellation design tools, we focus on the all important decoding problem. In this section, we develop efficient algorithms for decoding our proposed new codes. Interestingly, such decoding algorithms are also applicable to existing codes. For ease of presentation, we first focus on the case that the constellation has only one block (i.e.,  $b = 0$ ) and the receiver is equipped with only one antenna (i.e.,  $N = 1$ ). Subsequently, we discuss the decoding for the general cases of multiple blocks and multiple receiver antennas (i.e.,  $b \geq 0$  and  $N \geq 1$ ).

When  $b = 0$ , the constellation reduces to the continuous diagonal code. The signal constellation consists of  $L$  diagonal matrices  $V_\ell = \Lambda^\ell$ ,  $\ell = 0, 1, \dots, L-1$ . For  $N = 1$ , the received signal  $X_\tau \in \mathbb{C}^{M \times 1}$  is a complex vector. As described in [5], the ML decoding problem can be reformulated as a problem of minimizing a Euclidean norm as follows:

$$\begin{aligned} \hat{z}_\tau^{\text{ML}} &= \arg \min_{\ell} \|X_\tau - V_\ell X_{\tau-1}\|_F^2 \\ &\approx \arg \min_{\ell} \sum_{m=1}^M [(C_m \lambda_m \ell - C_m \varphi_m) \bmod^* C_m L]^2 \end{aligned} \quad (8)$$

where

$$C_m = \sqrt{|[X_\tau]_{m1} [X_{\tau-1}]_{m1}|}, \quad \varphi_m = \arg \left( \frac{[X_\tau]_{m1}}{[X_{\tau-1}]_{m1}} \right) \frac{L}{2\pi}.$$

It has been demonstrated in [5] that the approximation in (8) is extremely accurate. Therefore, the decoding problem for the case  $b = 0$ ,  $N = 1$  has been transformed into the minimization problem of finding

$$\hat{z}^{\text{eucl}} = \arg \min_{\ell} \sum_{m=1}^M [(C_m \lambda_m \ell - C_m \varphi_m) \bmod^* C_m L]^2. \quad (9)$$

In the following sub-sections we develop an efficient algorithm for this minimization problem.

### 3.1 Lattice Decoding Algorithms

In the special case that  $\lambda_m$ ,  $m = 1, \dots, M$  are integers, the continuous diagonal code reduces to the cyclic group code. In order to decode the cyclic group code, Clarkson *et al.*, [5] developed an approximate solution for the minimization problem (9). The key steps are as follows:

1. Reformulate minimization problem (9) as a lattice closest point problem

$$\arg \min_{y \in \mathbb{Z}^{1 \times M}} \|yG - \xi\| \quad (10)$$

where  $\xi = [\xi_1, \dots, \xi_M]$  with  $\xi_m = C_m \varphi_m$ ,  $m = 1, \dots, M$  and  $G$  is a  $M \times M$  generator matrix such that

$$[G]_{pq} = \begin{cases} C_q \lambda_q & \text{for } p = 1 \text{ and } 1 \leq q \leq M, \\ C_q L & \text{for } 1 < p = q \leq M, \\ 0 & \text{else.} \end{cases}$$

2. Apply the “LLL” lattice algorithm [26] to find an approximate solution  $\tilde{y} = [\tilde{y}_1, \dots, \tilde{y}_M]$  for (10). An estimate for  $\hat{z}^{\text{eucl}}$  is taken as  $\tilde{y}_1 \bmod L$ .

While the “LLL” lattice algorithm approximately solves (10), existing sphere decoder algorithms (see, e.g., [2], [6, 8], [9], [48] and the references therein) can provide an exact solution for (10) and hence improve decoding accuracy. The sphere decoder takes advantage of the lattice structure of the received signals and proceeds as follows: (i) It searches the closest lattice points to the received signal which are enclosed in a sphere centered at the received signal; (ii) each time a lattice point of a smaller norm is found, it reduces the sphere radius accordingly and restart the search until an empty sphere is reached. The choice of initial radius depends on the lattice considered, as well as on the additive noise level. At the heart of the sphere decoder algorithm is the subroutine which serves the purposes of: (a) determining whether a sphere  $\|yG - \xi\|^2 < \gamma^2$  with fixed radius  $\gamma > 0$  is empty; (b) detecting a vector in it otherwise. A Cholesky factorization is performed to find an upper triangular matrix  $D$  so that  $D^\top D = GG^\top$ , from which the boundary conditions of the sphere can be derived as

$$z_k - \sqrt{\frac{\vartheta_k}{t_{kk}}} - \varpi_k < y_k < \sqrt{\frac{\vartheta_k}{t_{kk}}} - \varpi_k + z_k, \quad k = M, M-1, \dots, 1 \quad (11)$$

where

$$[z_1, \dots, z_M] = \xi G^{-1}, \quad \varpi_k = \sum_{j=k+1}^M t_{kj}(y_j - z_j), \quad k = 1, \dots, M$$

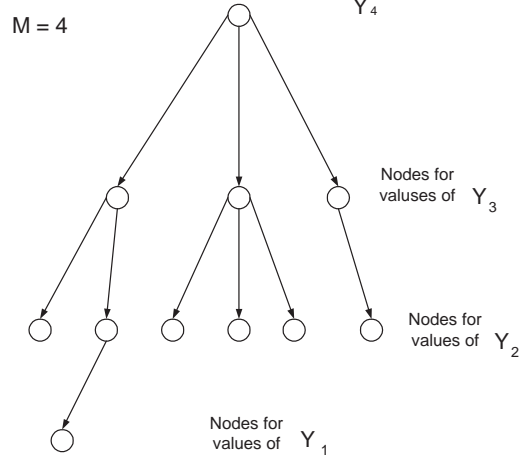


Figure 2: A tree representation of values of coordinates to be investigated for a fixed  $y_M$ .

and

$$\vartheta_M = \gamma^2, \quad \vartheta_{k-1} = \vartheta_k - t_{kk} (y_k - z_k + \varpi_k)^2, \quad k = 2, \dots, M$$

with  $t_{kk} = [D]_{kk}^2$ ,  $k = 1, \dots, M$  and  $t_{kj} = \frac{[D]_{kj}}{[D]_{kk}}$ ,  $1 \leq k < j \leq M$  (see, e.g., [9, 48] for details). Clearly, the boundary of  $y_k$  depends on values of  $y_j$ ,  $j = k + 1, \dots, M$ . If the set of feasible values of  $y_M$ , denoted by  $\mathcal{I}_M$ , is not empty, then for each member  $y_M$  of  $\mathcal{I}_M$  the values of other coordinates needed to be evaluated in the sphere decoder algorithm can be represented as the nodes of a tree starting from  $y_M$ . The following Figure 2 depicts this tree structure. In the tree, the children nodes are generated from the parent nodes in accordance with the boundary equation (11). A path of length  $M$  (i.e., consisting of  $M$  nodes) corresponds to a vector located in the sphere. When  $\mathcal{I}_M$  has multiple members, the task of the core subroutine is to search among the multiple trees to determine whether there is a path of length  $M$  and identify one if there exists. It should be noted that, in sphere decoding, *most of the computational efforts are devoted to the evaluation of paths of length less than  $M$ .*

### 3.2 Removing the Curse of Dimensionality

In the general case that  $\lambda_m$ ,  $m = 1, \dots, M$  are continuous parameters, the minimization problem in (9) lacks the lattice structure. Hence, existing sphere decoder algorithms and “LLL” lattice algorithm are not applicable. Moreover, even for the special case of cyclic group code, the lattice decoding algorithms described in the last subsection aim to solve a closest point problem of dimension  $M$  (the dimension will be expanded to  $MN$  when using  $N$  receiver antennas). The computational complexity may be too high when the number of transmitter antennas  $M$  (or the number of receiver antennas  $N$ ) is large. Therefore, it is crucial to reduce the dimension of the underlying closest point problem by further exploiting the diagonal structure of signal constellation. We achieve the reduction and improve efficiency with a new decoding algorithm, applicable to the general case that  $\lambda_m$ ,  $m = 1, \dots, M$  are continuous parameters. As a critical step to reduce decoding complexity, we will show next that the dimension of the related closest point problem can be reduced to one.

**Theorem 3** *Define*

$$\begin{aligned} \mathcal{S} = & \{(y_1, \dots, y_M) \mid y_1 \in \mathbb{Z} \\ & \text{and } -\frac{L}{2} + \varphi_1 \leq y_1 < \frac{L}{2} + \varphi_1; \\ & y_m = \lceil \frac{\varphi_m}{L} - (\frac{y_1}{L} - \lfloor \frac{y_1}{L} \rfloor) \lambda_m - \frac{1}{2} \rceil - \lfloor \frac{y_1}{L} \rfloor \lambda_m \\ & \text{for } m = 2, \dots, M\}. \end{aligned}$$

*Suppose that there exists a unique  $\hat{\ell} \in \{0, 1, \dots, L-1\}$  such that*

$$\begin{aligned} & \sum_{m=1}^M [(C_m \lambda_m \hat{\ell} - C_m \varphi_m) \bmod^* C_m L]^2 \\ = & \min_{\ell} \sum_{m=1}^M [(C_m \lambda_m \ell - C_m \varphi_m) \bmod^* C_m L]^2. \end{aligned}$$

*Then*

$$\begin{aligned} \hat{\ell} &= \arg \min_{\ell} \sum_{m=1}^M [(C_m \lambda_m \ell - C_m \varphi_m) \bmod^* C_m L]^2 \\ &= \hat{y}_1 - \left\lfloor \frac{\hat{y}_1}{L} \right\rfloor L \end{aligned}$$

where  $\hat{y}_1$  is first entry of

$$\hat{y} = [\hat{y}_1, \dots, \hat{y}_M] = \arg \min_{y \in \mathcal{S}} \|yG - \xi\|^2.$$

See Appendix C for a proof.

It can be seen from Theorem 3 that  $y_q$ , for  $q = 2, \dots, M$ , is uniquely determined by  $y_1$ . Hence, finding

$$\arg \min_{y \in \mathcal{S}} \|yG - \xi\|^2 \quad (12)$$

is essentially a one-dimensional closest point problem.

Next, by exploiting the special structure of the constellation, we derive extremely simple boundary conditions for the sphere  $\{y \in \mathcal{S} \mid \|yG - \xi\|^2 < \gamma^2\}$ .

**Theorem 4** *Let  $y = [y_1, \dots, y_M] \in \mathcal{S}$ . Define*

$$\mu_1 = [C_1 (y_1 - \varphi_1)]^2$$

*and*

$$\mu_m = \mu_{m-1} + [C_m (Ly_m + \lambda_m y_1 - \varphi_m)]^2$$

*for  $m = 2, \dots, M$ . Then  $\|yG - \xi\|^2 < \gamma^2$  if and only if  $y_1$  is an integer satisfying*

$$\varphi_1 - \frac{\gamma}{C_1} < y_1 < \frac{\gamma}{C_1} + \varphi_1, \quad -\frac{L}{2} + \varphi_1 \leq y_1 < \frac{L}{2} + \varphi_1 \quad (13)$$

*and*

$$\mu_m < \gamma^2 \text{ for } m = 2, \dots, M. \quad (14)$$

See Appendix D for a proof.

It can be seen that the conditions in (13) determine an interval  $\mathcal{I}_1$  of feasible values for  $y_1$ . For each value of  $y_1 \in \mathcal{I}_1$ , we only need to evaluate the simple conditions in (14). This is in sharp contrast to the search over a tree structure described above in the context of sphere decoder.

It should be noted that the sphere decoding algorithm is originally devised to find closest lattice points. In general, our decoding problem is not a problem of searching closest lattice points. However, we can still use the sphere decoding algorithm because the enumeration of interior points of a sphere can be efficiently done as the case of a lattice problem. Moreover, Theorem 4 indicates that the “sphere” can actually be reduced to an “interval” of one dimension.

### 3.3 Simplified Sphere Decoder

Using the reduction of dimensionality described in the last subsection, we now develop a new decoding algorithm which also applies to continuous diagonal codes, FPF codes  $G_{m,r}$ , non-group codes  $S_{m,s}$  and products of groups. To further enhance efficiency, we adopt the “zigzag” searching strategy originated in [37] and the idea proposed in [8] for avoiding repeated computations.

Obviously, the search for  $y_1$  can significantly affect the efficiency. Let  $\hat{y}$  denote the vector corresponding to the transmitted signal. Intuitively, for moderate and high SNR, it is more likely for the received signal  $\xi$  to be closer to  $\hat{y} G$ . Since  $|y_1 - \varphi_1| < \|yG - \xi\|$ , we should first investigate  $y_1$  which is closer to  $\varphi_1$  for a better chance of detecting  $\hat{y}$ . Therefore, we shall investigate  $y_1$  in the following sequence,

$$\lfloor \varphi_1 \rfloor + (-1)^k \left\lfloor \frac{k}{2} \right\rfloor, \quad k = 0, 1, 2, \dots \quad (15)$$

That is, the investigation is started from  $\lfloor \varphi_1 \rfloor$  and proceeded in a “zigzag” order in the outward directions (see, e.g., [2], [37]). Note that condition (13) implies

$$-\left\lceil \min\left(\frac{\gamma}{C_1}, \frac{L}{2}\right) \right\rceil \leq y_1 - \lfloor \varphi_1 \rfloor \leq \left\lceil \min\left(\frac{\gamma}{C_1}, \frac{L}{2}\right) \right\rceil.$$

Hence, it suffices to investigate

$$y_1 = \lfloor \varphi_1 \rfloor + (-1)^k \left\lfloor \frac{k}{2} \right\rfloor, \quad k = 0, 1, \dots, 2 \left\lceil \min\left(\frac{\gamma}{C_1}, \frac{L}{2}\right) \right\rceil.$$

It is also important to avoid repeated investigation of  $y_1$ . When a value of  $y_1$  is found to satisfy condition (14), the radius  $\gamma$  is reduced as  $\sqrt{\mu_M}$  and the interval confining  $y_1$  is consequently shrunk. In this way, the range of  $y_1$  needed to be investigated is squeezed from outside. To improve efficiency, we use the idea of [8] to ensure that the range of  $y_1$  needed to be investigated is also squeezed from inside. The idea is based on the following observation:

*For a given radius  $\gamma$ , if a value of  $y_1$  violates the boundary conditions, then the same value of  $y_1$  also violates the corresponding boundary conditions after  $\gamma$  is reduced.*

Therefore, we can keep a record for the values of  $y_1$  which have been investigated in order to avoid repeated computation. For this purpose, the index variable  $k$  in (15) can be used as an indicator for the range of values investigated.

In summary, the decoding algorithm is presented as follows.

STEP 1. Input  $\gamma \leftarrow \gamma_{\text{init}}$  where initial radius  $\gamma_{\text{init}}$  is chosen based on noise level. Let  $k \leftarrow 0$  and  $\tilde{y}_1 \leftarrow \lfloor \varphi_1 \rfloor$ .

STEP 2. Let  $k_{\text{max}} \leftarrow 2 \left\lceil \min \left( \frac{\gamma}{C_1}, \frac{L}{2} \right) \right\rceil$ .

STEP 3. If  $k \leq k_{\text{max}}$ , let  $y_1 \leftarrow \lfloor \varphi_1 \rfloor + (-1)^k \lfloor \frac{k}{2} \rfloor$  and  $k \leftarrow k + 1$ . Otherwise, let  $\gamma \leftarrow \frac{4}{3}\gamma$ ,  $k \leftarrow 0$  and go to STEP 2.

STEP 4. If condition (14) is violated, go to STEP 3. Otherwise, let  $\tilde{y}_1 \leftarrow y_1$ ,  $\gamma \leftarrow \sqrt{\mu_M}$ .

STEP 5. Using  $\gamma$ ,  $\tilde{y}_1$  and  $k$  as input, call subroutine CLOSEST-POINT to find  $\hat{y}_1$ . Then  $\hat{z}^{\text{eucl}}$  is calculated as  $\hat{z}^{\text{eucl}} = \hat{y}_1 - \left\lfloor \frac{\hat{y}_1}{L} \right\rfloor L$ . Return  $\hat{z}^{\text{eucl}}$  as the estimate and stop.

The subroutine CLOSEST-POINT is presented as follows.

Function: CLOSEST-POINT

STEP 1. Input  $\gamma$ ,  $\tilde{y}_1$  and  $k$ . Let  $\hat{y}_1 \leftarrow \tilde{y}_1$ .

STEP 2. Let  $k_{\text{max}} \leftarrow 2 \left\lceil \min \left( \frac{\gamma}{C_1}, \frac{L}{2} \right) \right\rceil$ .

STEP 3. If  $k \leq k_{\text{max}}$ , let  $y_1 \leftarrow \lfloor \varphi_1 \rfloor + (-1)^k \lfloor \frac{k}{2} \rfloor$  and  $k \leftarrow k + 1$ . Otherwise, go to STEP 5.

STEP 4. If condition (14) is satisfied, then let  $\hat{y}_1 \leftarrow y_1$ ,  $\gamma \leftarrow \sqrt{\mu_M}$  and go to STEP 2. Otherwise, go to STEP 3.

STEP 5. Return  $\hat{y}_1$  and stop.

It can be seen from STEP 3 that the index  $k$  has served the purpose of avoiding repeated investigations. Once a nonempty sphere is detected, no value of  $y_1$  is investigated more than once among the subsequent smaller spheres.

It should be also noted that, compared to conventional sphere decoder algorithms, many computationally expensive steps have been avoided in our algorithm. For examples, the Cholesky factorization of  $GG^\top$  and the computation of  $\xi G^{-1}$  are not needed in our algorithms.

### 3.4 Sphere Decoding – The General Case

We now discuss the decoding problem for the general case of multiple receiver antennas and multiple block constellation (i.e.,  $N \geq 1$ ,  $b \geq 0$ ). Since the ML-decoding is computationally difficult, our goal is to develop a sub-optimal decoding method with low decoding complexity. Note that the ML decoding finds

$$\arg \min_{\ell, q} \|X_\tau - A_q \Lambda_q^\ell B_q X_{\tau-1}\|_F^2, \quad (16)$$

which can be done by obtaining

$$\arg \min_{\ell} \|X_\tau - A_q \Lambda_q^\ell B_q X_{\tau-1}\|_F^2 \quad (17)$$

for  $q = 0, 1, \dots, 2^b - 1$  and seeking the tuple  $(\hat{q}, \hat{\ell})$  minimizing the Frobenius norm. This method is of sequential nature and has been used in [39] for decoding FPF code  $G_{m,r}$ , non-group code  $S_{m,s}$  and products of groups with the “LLL” lattice algorithm sequentially applied to solve (17).

In the general case the underlying closest point problem is of dimension  $MN$  and sequential approaches are inefficient. We use the simplified sphere decoder algorithm developed in the previous subsection and transform the decoding problem into  $2^b$  one-dimensional closest point problems that can be solved in parallel.

Since the Frobenius norm of a matrix is invariant under unitary transformations, we have

$$\begin{aligned} \|X_\tau - A_q \Lambda_q^\ell B_q X_{\tau-1}\|_F^2 &= \|A_q^\dagger (X_\tau - A_q \Lambda_q^\ell B_q X_{\tau-1})\|_F^2 \\ &= \|A_q^\dagger X_\tau - \Lambda_q^\ell B_q X_{\tau-1}\|_F^2. \end{aligned}$$

By a similar method as that of [5], we can show that

$$\begin{aligned} & \|A_q^\dagger X_\tau - \Lambda_q^\ell B_q X_{\tau-1}\|_F^2 \\ &= \sum_{n=1}^N \sum_{m=1}^M \left| [A_q^\dagger X_\tau]_{mn} - e^{i2\pi\lambda_{q,m}\ell/L} [B_q X_{\tau-1}]_{mn} \right|^2 \\ &= \|X_\tau\|_F^2 + \|X_{\tau-1}\|_F^2 \\ &\quad - 2 \sum_{n=1}^N \sum_{m=1}^M C_{m,n}^2 \cos([( \lambda_{q,m} \ell - \varphi_{m,n} ) \bmod^* L] 2\pi/L) \\ &\approx \|X_\tau\|_F^2 + \|X_{\tau-1}\|_F^2 - 2 \sum_{n=1}^N \sum_{m=1}^M C_{m,n}^2 \end{aligned}$$



$$\begin{aligned}
& + \sum_{n=1}^N \sum_{m=1}^M C_{m,n}^2 ([(\lambda_{q,m} \ell - \varphi_{m,n}) \bmod^* L] 2\pi/L)^2 \\
& = \frac{4\pi^2}{L^2} \Delta_q + \frac{4\pi^2}{L^2} \times \\
& \quad \sum_{n=1}^N \sum_{m=1}^M [(C_{m,n} \lambda_{q,m} \ell - C_{m,n} \varphi_{m,n}) \bmod^* C_{m,n} L]^2
\end{aligned} \tag{18}$$

where

$$\begin{aligned}
C_{m,n} &= \sqrt{|[A_q^\dagger X_\tau]_{mn} [B_q X_{\tau-1}]_{mn}|}, \\
\varphi_{m,n} &= \arg \left( \frac{[A_q^\dagger X_\tau]_{mn}}{[B_q X_{\tau-1}]_{mn}} \right) \frac{L}{2\pi},
\end{aligned}$$

and

$$\Delta_q = \frac{L^2 \|\text{abs}(A_q^\dagger X_\tau) - \text{abs}(B_q X_{\tau-1})\|_F^2}{4\pi^2}.$$

Define a  $MN \times MN$  matrix  $G^q$  such that

$$[G^q]_{kj} = \begin{cases} C_{(j - \lfloor \frac{j-1}{M} \rfloor M, \lfloor \frac{j-1}{M} \rfloor + 1)} \lambda_{(q, j - \lfloor \frac{j-1}{M} \rfloor M)} & \text{for } k = 1 \text{ and } 1 \leq j \leq MN; \\ C_{(j - \lfloor \frac{j-1}{M} \rfloor M, \lfloor \frac{j-1}{M} \rfloor + 1)} L & \text{for } 1 < k = j \leq MN; \\ 0 & \text{else.} \end{cases}$$

Define a row vector  $\xi^q = [\xi_1, \dots, \xi_{MN}]$  such that

$$\xi_k = C_{(k - \lfloor \frac{k-1}{M} \rfloor M, \lfloor \frac{k-1}{M} \rfloor + 1)} \varphi_{(k - \lfloor \frac{k-1}{M} \rfloor M, \lfloor \frac{k-1}{M} \rfloor + 1)}$$

for  $k = 1, \dots, MN$ . Define  $\psi_k = \varphi_{(k - \lfloor \frac{k-1}{M} \rfloor M, \lfloor \frac{k-1}{M} \rfloor + 1)}$  and  $\beta_k = \lambda_{(q, \lfloor \frac{k-1}{M} \rfloor + 1)}$  for  $k = 1, \dots, MN$ . Define

$$\begin{aligned}
\mathcal{S}^q &= \{(y_1, \dots, y_{MN}) \mid y_1 \in \mathbb{Z} \\
&\quad \text{and } -\frac{L}{2} + \psi_1 \leq y_1 < \frac{L}{2} + \psi_1; \\
&\quad y_k = \left\lceil \frac{\psi_k}{L} - (\frac{y_1}{L} - \lfloor \frac{y_1}{L} \rfloor) \beta_k - \frac{1}{2} \right\rceil - \lfloor \frac{y_1}{L} \rfloor \beta_k \\
&\quad \text{for } k = 2, \dots, MN\}.
\end{aligned}$$

Then, by Theorem 3, we have

$$\begin{aligned} & \min_{\ell} \sum_{n=1}^N \sum_{m=1}^M [(C_{m,n} \lambda_{q,m} \ell - C_{m,n} \varphi_{m,n}) \bmod^* C_{m,n} L]^2 \\ &= \min_{y \in \mathcal{S}^q} \|yG^q - \xi^q\|^2. \end{aligned} \quad (19)$$

It follows from (18) and (19) that

$$\begin{aligned} & \|A_q^\dagger X_\tau - \Lambda_q^\ell B_q X_{\tau-1}\|_{\mathbb{F}}^2 \\ & \approx \frac{4\pi^2}{L^2} \min_{y \in \mathcal{S}^q} (\|yG^q - \xi^q\|^2 + \Delta_q), \end{aligned}$$

leading to

$$\begin{aligned} & \min_{\ell, q} \|X_\tau - A_q \Lambda_q^\ell B_q X_{\tau-1}\|_{\mathbb{F}}^2 \\ & \approx \frac{4\pi^2}{L^2} \min_q \min_{y \in \mathcal{S}^q} (\|yG^q - \xi^q\|^2 + \Delta_q). \end{aligned}$$

Hence, by Theorem 3, the maximum likelihood decoder can be well approximated by

$$(\hat{q}, \hat{\ell}) = \left( \hat{q}, \hat{y}_1 - \left\lfloor \frac{\hat{y}_1}{L} \right\rfloor \right)$$

where  $\hat{y}_1$  is the first entry of  $\hat{y} = [\hat{y}_1, \dots, \hat{y}_{MN}]$  such that

$$(\hat{q}, \hat{y}) = \arg \min_q \min_{y \in \mathcal{S}^q} (\|yG^q - \xi^q\|^2 + \Delta_q).$$

The above analysis shows that the efficiency of the decoding problem (16) can be enhanced by sequentially applying the simplified sphere decoder algorithms developed in the last subsection. In the next sub-section we shall improve the efficiency even further by developing a parallel search strategy.

### 3.5 Parallel Sphere Decoding

The sequential sphere decoder algorithm introduced in the last subsection involves  $2^b$  independent sphere decoding processes. When the constellation consists of many blocks (i.e., large  $2^b$ ), the sequential decoding may be too time consuming, but given the independence of each sphere decoding all searches can be executed in parallel. Specifically, since it has been shown

in the previous subsection that the ML decoding problem (16) can be reformulated as the sub-optimal decoding problem

$$\arg \min_{q=0,1,\dots,2^b-1} \min_{y \in \mathcal{S}^q} (||yG^q - \xi^q||^2 + \Delta_q),$$

we can apply in parallel the simplified sphere decoder algorithm to investigate the following  $2^b$  spheres:

$$\{y \in \mathcal{S}_q \mid ||yG^q - \xi^q||^2 < \gamma_q^2\}, \quad q = 0, 1, \dots, 2^b - 1$$

where

$$\gamma_q^2 = \gamma^2 - \Delta_q, \quad q = 0, 1, \dots, 2^b - 1 \quad (20)$$

with parameter  $\gamma$  controlling the sizes of all spheres. The choice of the initial value of  $\gamma$  is similar to choosing the initial radius of conventional sphere decoder. For a fixed value of  $\gamma$ , the  $2^b$  spheres respectively determine  $2^b$  sets of feasible  $y_1$  values based on (13). Let the set for the  $q$ -th sphere be denoted as  $\mathcal{I}^q$ . We investigate the  $y_1$  values of these sets in a round robin order. That is, the sets are visited in the following sequence

$$\mathcal{I}^0, \mathcal{I}^1, \dots, \mathcal{I}^{2^b-1}; \mathcal{I}^0, \mathcal{I}^1, \dots, \mathcal{I}^{2^b-1}; \dots\dots\dots$$

Of course, any value of  $y_1$  will be eliminated from its corresponding set after evaluation. Once a value of  $y_1$  from set  $\mathcal{I}^q$  is found to guarantee (13) and (14),  $\gamma_q$  is reduced as  $\sqrt{\mu_M}$ . Subsequently,  $\gamma$  is reduced as  $\sqrt{\gamma_q^2 + \Delta_q}$  and the radius of other spheres are decreased accordingly by (20). When no value of  $y_1$  from any set satisfies (13) and (14),  $\gamma$  will be increased and consequently all the spheres are enlarged based on (20). All the spheres keep enlarging before detecting a value of  $y_1$  guaranteeing (13) and (14). Once the value of  $y_1$  is found, all spheres begin to shrink. The shrinking process is very quick due to the parallel mechanism. This process is terminated when all these sets become empty. The solution of decoding problem (16) is given as the tuple

$$\left( \hat{q}, \hat{y}_1 - \left\lfloor \frac{\hat{y}_1}{L} \right\rfloor L \right)$$

where  $\hat{y}_1 \in \mathcal{I}^{\hat{q}}$  and  $\hat{y}_1$  is last value found to guarantee (13) and (14). It should be noted that, in this decoding process, *only one CPU processor is needed*.

## 4 Illustrative Examples

In this paper, we only design constellations for the special structure that  $\Lambda_q = \Lambda$ ,  $A_q = I$  for  $q = 0, \dots, 2^b - 1$ . The computational effort has been significantly reduced by applying Theorem 1. Better codes (with lower bit error rate but equivalent decoding complexity) can be obtained if we allow general  $A_q$  and  $\Lambda_q$ . However, the searching time will be substantially increased if the constellation size is large. Even in this limited case, the new design paradigm generates unitary space-time constellations which significantly outperform existing ones. In the following, we show the simulation results of our codes as compared to existing codes. In comparison of the bit error rate performance, we have used the Gray code bit mapping for the orthogonal design and CD codes, and the binary-to-decimal conversion mapping for our codes, cyclic group codes, FPF codes and product of groups. The data of our unitary space-time codes are reported in Appendix E. The details of orthogonal designs we used in our simulation is provided in Appendix F.

For the case of two transmit antennas and one receiver antenna, our computational experience indicates that it is hard to achieve significant performance improvement upon the orthogonal design proposed in [44]. By using nonconstant modulus constellations, the performance of [24] further improves upon that of [44] at the price of the complexity of estimating the channel power and signal power. However, when the number of transmit antennas is more than two or the number of receiver antennas is more than one, the differential detection scheme based on orthogonal designs subjects to significant performance loss.

We compared the performance of our code with the differential detection schemes using orthogonal designs in Figures 3-7. In general, our codes significantly outperform orthogonal designs at the price of relatively higher decoding complexity. It can be seen from Figure 3 that, with spectral efficiency  $R = 6$  bits per channel use, our code (with block number 16, i.e.,  $b = 4$ ) improves upon orthogonal design over 10 dB at block error rate  $10^{-1}$  when using two transmitter antennas and two receiver antennas. It is shown in Figure 4 that, with spectral efficiency  $R = 4$  bits per channel use, our code improves upon orthogonal design about 11 dB at block error rate  $2 \times 10^{-2}$  when using 3 transmitter antennas and one receiver antenna. It can be seen from Figure 6 that, with spectral efficiency  $R = 3$  bits per channel use, our code improves upon orthogonal design about 6 dB at bit error rate  $10^{-3}$  when using 4 transmitter antennas and 2 receiver antennas. These examples demonstrate that orthogonal designs suffer from substantial

performance penalty. Such penalty becomes more severe when using multiple receiver antennas, or using more than two transmit antennas, or operating at high spectral efficiency.

We compared the performance of our codes with Caley differential codes in Figures 3-7. Figure 3 shows that, with spectral efficiency  $R = 6$  bits per channel use, our code (with block number 16, i.e.,  $b = 4$ ) improves upon Caley differential code (reported in page 1495 of [16]), about 9 dB at block error rate  $6 \times 10^{-2}$  when using two transmitter antennas and two receiver antennas. The improvements of our codes with block number 4 and 8 are respectively 4 dB and 7 dB at block error rate  $6 \times 10^{-2}$ . The data of CD codes we used in simulation for Figures 4-7 is not available in the literature. We followed the design method proposed in [16] to search the corresponding CD codes. As described in [16], the performance metric used in the optimization is the average logarithm determinant  $\xi(\mathcal{V})$ . The number of data streams  $Q$  should be chosen as large as possible under constraint (30) of [16]. For a given spectral efficiency  $R$ , once  $Q$  is fixed, the set  $\mathcal{A}_r$  for  $\{\alpha_q\}$  is determined and is provided in [16]. We obtained CD codes via extensive gradient-based optimization. The values of tuple  $(Q, \xi)$  for the CD codes corresponding to Figures 4-7 are, respectively,  $(4, 0.2610)$ ,  $(8, 0.5832)$ ,  $(12, 0.3619)$  and  $(12, 0.5401)$ . As can be seen from Figures 4-7, the performance of CD codes is not comparable with that of our codes. However, we can see that CD codes are generally better than cyclic group codes and orthogonal designs in terms of bit (or block) error rate performance.

In Figures 4-6, we compared the performance of our proposed codes with the FPF codes proposed in [39]. It is seen from Figure 4 that, with spectral efficiency  $R = 4$  bits per channel use, our code (with block number 16, i.e.,  $b = 4$ ) improves upon the product of cyclic groups (see Table IV of [39]) about 2 dB at block error rate  $2 \times 10^{-3}$  when using 3 transmitter antennas and one receiver antenna. In Figure 5, our code slightly outperforms the product of groups code. However, our code has a lower decoding complexity since our code involves only 4 branches of sphere decoding, while the product of groups code involves 17 branches of sphere decoding. In this case, the  $T$  matrix is not available from [39]. We used the same diagonal elements,  $u = [1 \ 3 \ 4 \ 11]$ , as that of [39]. We searched the best  $T$  matrix based on the conventional criterion of diversity product maximization. We obtained a  $T$  matrix so that the constellation has diversity product 0.3118, which is greater than the previously known value, 0.3105, reported in [39]. In Figure 6, our code significantly outperforms the product of groups code. Our code with  $b = 4$  improves upon the product of groups code about 3 dB at bit error rate  $10^{-4}$ . Moreover, our code has a lower decoding complexity since

our code involves only 16 branches of sphere decoding, while the product of groups code involves 65 branches of sphere decoding. In this case, we used the same diagonal elements,  $u = [1, 14, 21, 34]$ , as that of [39]. We searched the best  $T$  matrix based on the conventional criterion of diversity product maximization. We obtained a  $T$  matrix so that the constellation has diversity product 0.1563, which is greater than the previously known value, 0.1539, reported in [39].

We compared the performance of our proposed code with cyclic group codes in Figures 3, 5 and 6. It is demonstrated that our codes significantly outperform cyclic group codes. For example, Figure 5 shows that, with spectral efficiency  $R = 2$  bits per channel use, our code (with block number 4, i.e.,  $b = 2$ ) improves upon the best previously known cyclic group code  $u = [1\ 25\ 97\ 107]$  (see Table I of [20]) about 3 dB at bit error rate  $10^{-3}$  when using 4 transmitter antennas and 2 receiver antennas. The cyclic group code corresponding to Figure 3 is  $u = [1, 1731]$  of diversity product 0.0265. The cyclic group code corresponding to Figure 6 is  $u = [1, 301, 1561, 1829]$  of diversity product 0.1035. We obtained these two cyclic group codes based on the conventional criterion of diversity product maximization.

Specially, we have presented continuous diagonal codes for many combinations of antenna numbers and constellation sizes in Table 1 of Appendix E. These continuous diagonal codes outperform cyclic group codes in terms of bit error rate. For example, in Figure 7, with spectral efficiency  $R = 2$  bits per channel use, our continuous diagonal code  $\Lambda = [1\ 11.8659\ 404.3640\ 592.2112\ 1328.7582\ 1489.9040]$  improves upon the best previously known cyclic group code  $u = [1\ 599\ 623\ 1445\ 1527\ 1715]$  (see Table I of [39]) about 1.5 dB at bit error rate  $10^{-4}$  when using 6 transmitter antennas and 2 receiver antennas. In Figure 7, it shown that our continuous diagonal code also substantially outperforms the orthogonal design and the CD code (with  $(Q, \xi) = (12, 0.5401)$  as mentioned before). However, the product of groups code has much better performance than our continuous diagonal code. For the product of groups code, we used the same diagonal elements,  $u = [1, 9, 21, 51, 53, 57]$ , as that of [39]. We searched the best  $T$  matrix based on the conventional criterion of diversity product maximization. We obtained a  $T$  matrix so that the constellation has diversity product 0.2098, which is greater than the previously known value, 0.2084, reported in [39].

It is important to note that, since the constellation size of many types of FPF codes is not a power of 2, the bit assignment is not trivial and may significantly increase bit error rate. The first method of bit assignment is to truncate the constellation as a smaller one so that the size is of a power of

2. The drawback with this mapping method is that a large portion of the signal matrices may be wasted. For example, suppose we have an optimal (or near optimal) constellation of 240 signal matrices but only 128 of them is used to convey information. One can argue that it may be better to directly seek the optimal (or near optimal) constellation of 128 signal matrices. The second method is to map  $n$  consecutive bits into  $m$  consecutively transmitted matrices where  $m$  and  $n$  are integers large enough so that  $2^n$  is close to the  $m$ -th power of constellation size (see, pp. 2356-2357 of [39]). Unfortunately, the bit error rate will be increased as the product of the block error rate and  $\eta m$  where  $\eta \in (0, 1)$  may not be small. Moreover, the decoding delay is increased as  $mM$  symbol periods, which may be intolerable for large  $m$  and  $M$ . The increase of bit error rate and decoding delay can be substantial since the factor  $m$  can be quite large. For example, when the constellation size is 240, the minimal values of integer  $m$  to guarantee  $1 < \frac{240^m}{2^n} \leq 1.05$  and  $1 < \frac{240^m}{2^n} \leq 1.01$  are respectively 10 and 118. It can be seen from the above analysis that for practical purpose the size of signal constellation should be a power of 2. This is one of the reasons why we choose code parameters to be continuous so that we can find unitary space-time constellations of any size.

Finally, we would like to point out that some of the constellations we obtained have *zero* diversity product. However, these constellations significantly outperform other constellations with much larger diversity product. Such constellations can be found in Appendix E for the following combinations: (i)  $M = N = 2$ ,  $b = 2$ ; (ii)  $M = N = 2$ ,  $b = 3$ ; (iii)  $M = 4$ ,  $N = 2$ ,  $b = 2$ . As comparing to diversity product, our computational experiments indicate that the trapezoid criterion introduced in Section 2 works quite well even in low SNR region.

## 5 Conclusion

We have proposed a new class of differential unitary space-time codes which has high performance, low encoding and decoding complexity. We have established a parallel sphere decoder algorithm which efficiently decodes our proposed code and existing codes such as cyclic group code, FPF code  $G_{m,r}$ , non-group code  $S_{m,s}$  and products of groups. We have proposed a new design

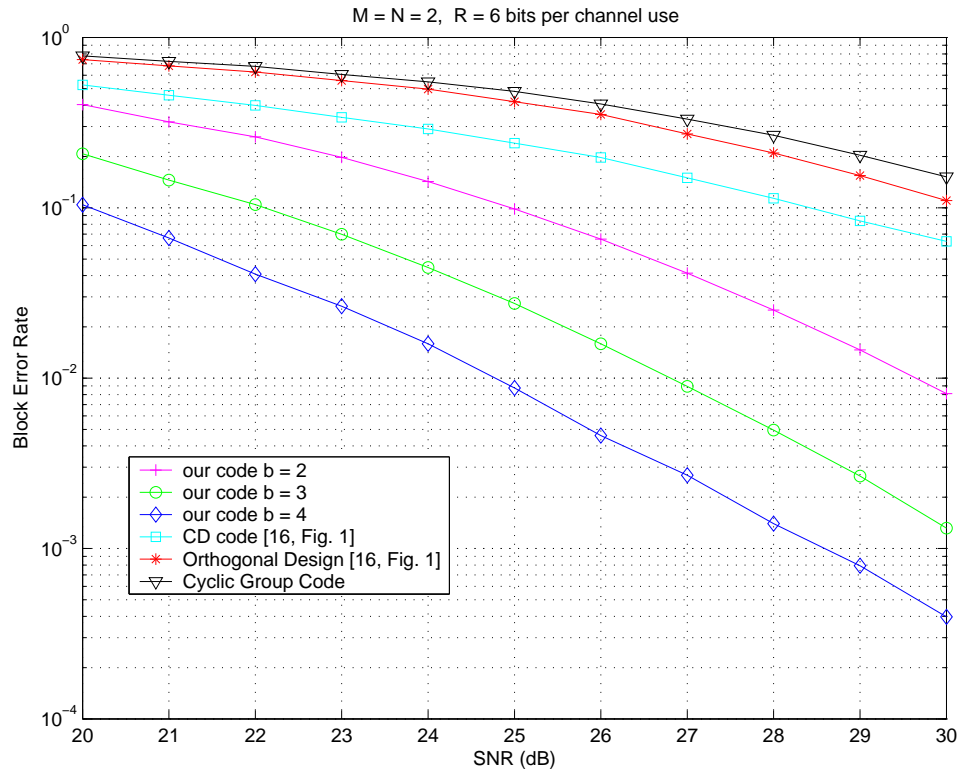


Figure 3: Performance simulations of constellations



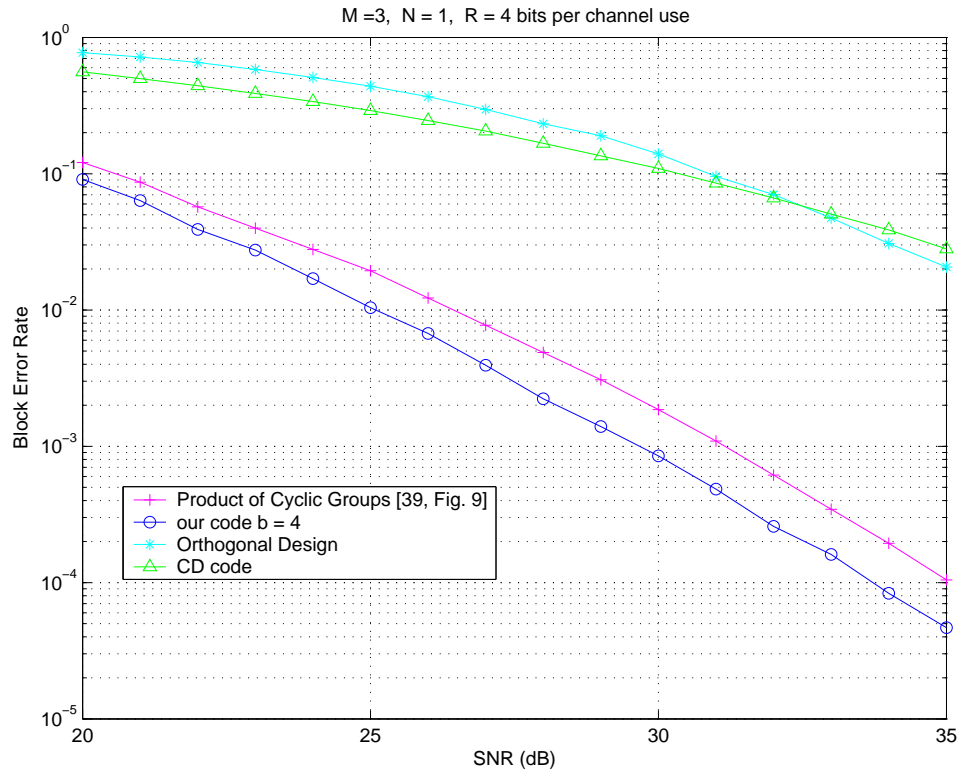


Figure 4: Performance simulations of constellations

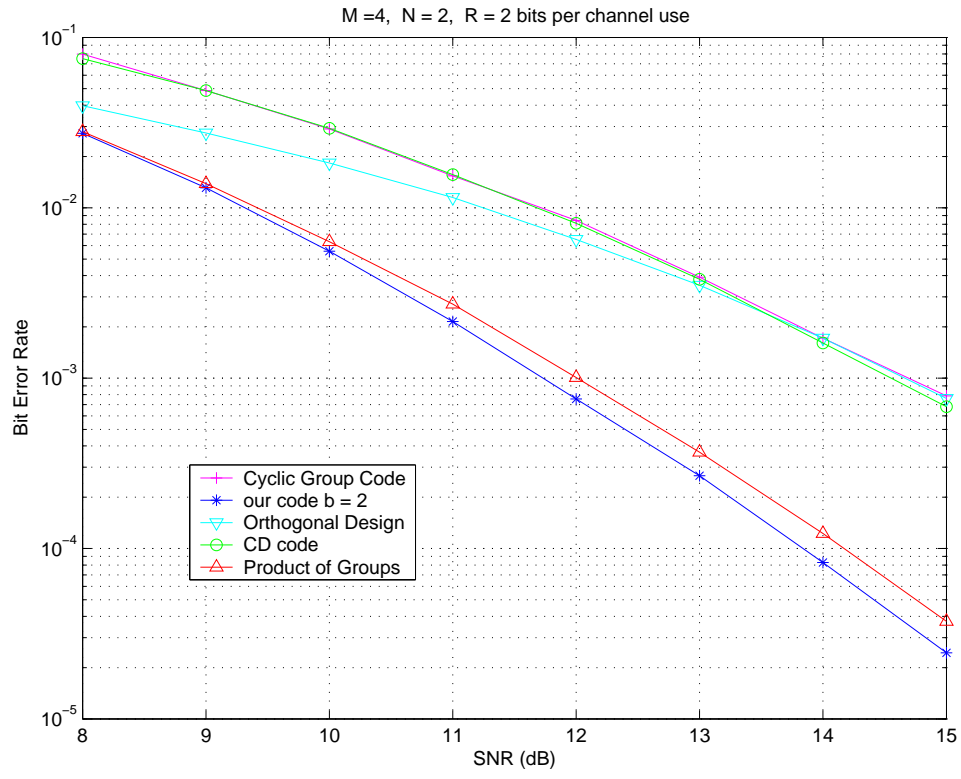


Figure 5: Performance simulations of constellations

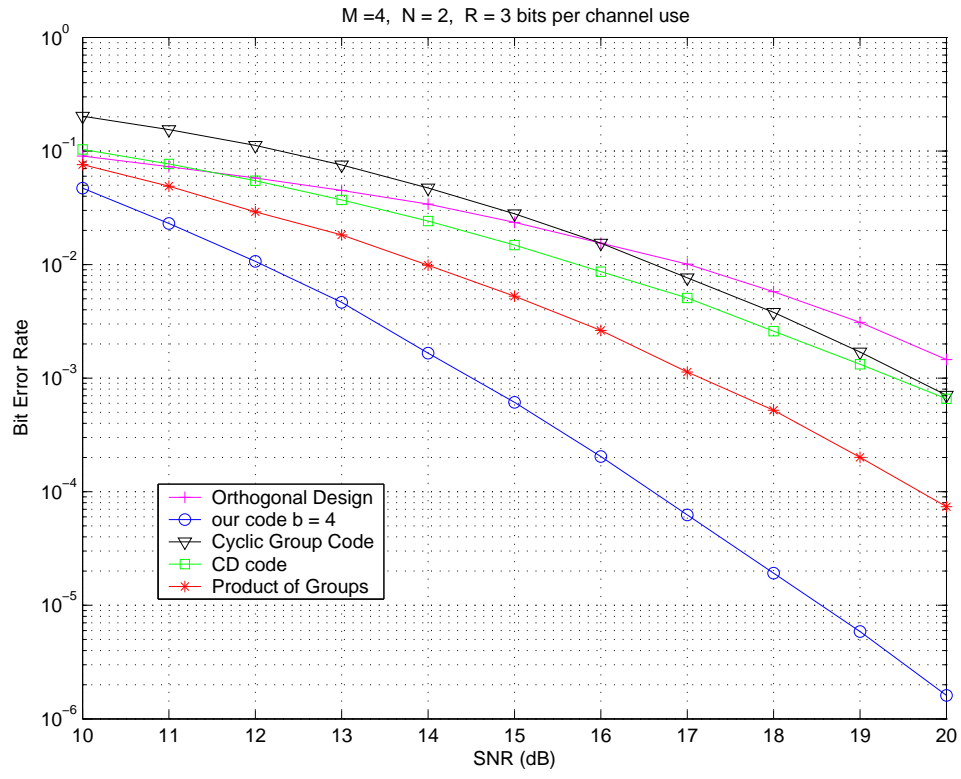


Figure 6: Performance simulations of constellations

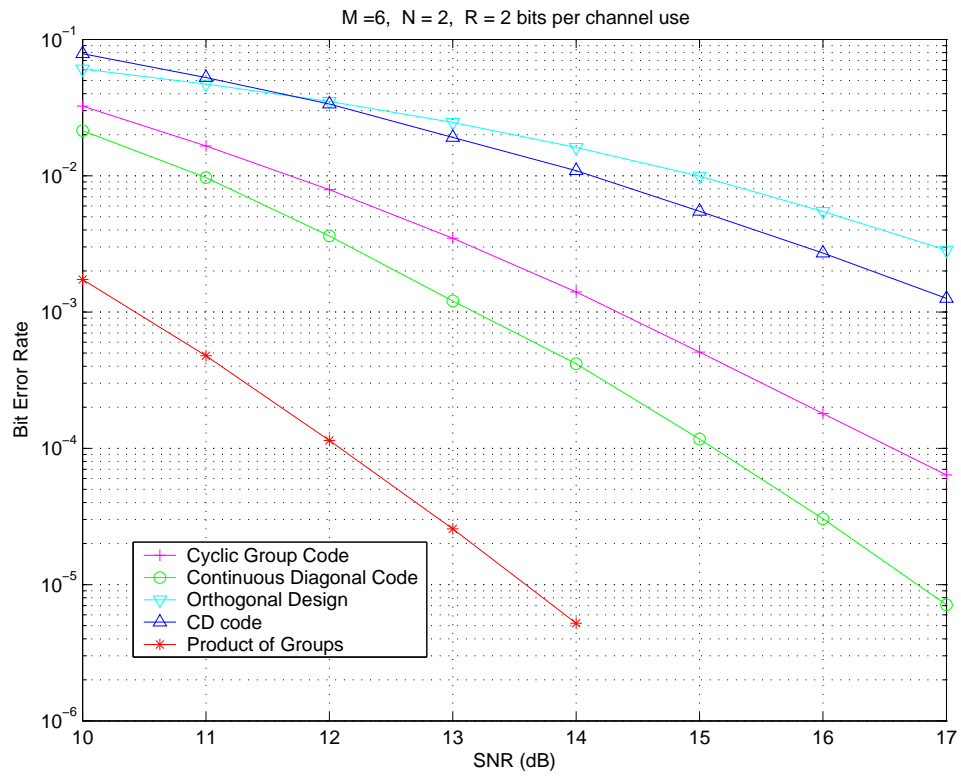


Figure 7: Performance simulations of constellations

criterion and powerful optimization techniques for designing unitary space-time codes. We have obtained constellations which significantly improve upon constellations reported in the literature.

## A PROOF OF THEOREM 1

From the illustration after Theorem 2, we see that, the Chernoff bound of the pair-wise error probability is invariant under unitary transforms. By such invariant property, we have

$$P(\Lambda_p^\ell, \Lambda_p^{\ell'}) = P(I, \Lambda^{\ell' - \ell}), \quad 0 \leq p \leq 2^b - 1.$$

Note that

$$d^H(p, p, \ell, \ell') = d^H(\ell, \ell')$$

for  $0 \leq p \leq 2^b - 1$ . Hence

$$\begin{aligned} & \sum_{p=0}^{2^b-1} \sum_{\ell=0}^{L-2} \sum_{\ell'=\ell+1}^{L-1} d^H(p, p, \ell, \ell') P(\Lambda_p^\ell, \Lambda_p^{\ell'}) \\ &= 2^b \sum_{\ell=0}^{L-2} \sum_{\ell'=\ell+1}^{L-1} d^H(\ell, \ell') P(I, \Lambda^{\ell' - \ell}) \\ &= 2^b \sum_{k=1}^{L-1} \sum_{\substack{\ell' - \ell = k \\ 0 \leq \ell \leq L-2 \\ \ell' \leq L-1}} d^H(\ell', \ell) P(I, \Lambda^k). \end{aligned} \quad (21)$$

It can be verified that

$$\begin{aligned} \sum_{\substack{\ell' - \ell = k \\ 0 \leq \ell \leq L-2 \\ \ell' \leq L-1}} d^H(\ell', \ell) &= \sum_{0 \leq \ell \leq L-k-1} d^H(\ell + k, \ell) \\ &= w(k). \end{aligned} \quad (22)$$

By (21) and (22),

$$\begin{aligned} & \sum_{p=0}^{2^b-1} \sum_{\ell=0}^{L-2} \sum_{\ell'=\ell+1}^{L-1} d^H(p, p, \ell, \ell') P(\Lambda_p^\ell, \Lambda_p^{\ell'}) \\ &= 2^b \sum_{k=1}^{L-1} w(k) P(I, \Lambda^k). \end{aligned} \quad (23)$$

Observing that

$$d^H(p, q, \ell, \ell') = d^H(\ell, \ell') + d^H(p, q)$$

and

$$P(A_p \Lambda_p^\ell B_p, A_q \Lambda_q^{\ell'} B_q) = P(B_p, \Lambda^{\ell' - \ell} B_q),$$

we have

$$\begin{aligned}
& \sum_{\ell=0}^{L-1} \sum_{\ell'=0}^{L-1} d^H(p, q, \ell, \ell') P(A_p \Lambda_p^\ell B_p, A_q \Lambda_q^{\ell'} B_q) \\
= & \sum_{\ell=0}^{L-1} \sum_{\ell'=0}^{L-1} d^H(\ell, \ell') P(B_p, \Lambda^{\ell' - \ell} B_q) \\
& + \sum_{\ell=0}^{L-1} \sum_{\ell'=0}^{L-1} d^H(p, q) P(B_p, \Lambda^{\ell' - \ell} B_q) \\
= & \sum_{k=-L+1}^{L-1} \sum_{\substack{\ell' - \ell = k \\ 0 \leq \ell \leq L-1 \\ 0 \leq \ell' \leq L-1}} d^H(\ell, \ell') P(B_p, \Lambda^k B_q) \\
& + d^H(p, q) \sum_{k=-L+1}^{L-1} P(B_p, \Lambda^k B_q). \tag{24}
\end{aligned}$$

Making use of symmetry, we can show that

$$\sum_{\substack{\ell' - \ell = k \\ 0 \leq \ell \leq L-1 \\ 0 \leq \ell' \leq L-1}} d^H(\ell, \ell') = w(|k|). \tag{25}$$

By (24) and (25),

$$\begin{aligned}
& \sum_{\ell=0}^{L-1} \sum_{\ell'=0}^{L-1} d^H(p, q, \ell, \ell') P(A_p \Lambda_p^\ell B_p, A_q \Lambda_q^{\ell'} B_q) \\
= & \sum_{k=-L+1}^{L-1} [w(|k|) + d^H(p, q)] P(B_p, \Lambda^k B_q). \tag{26}
\end{aligned}$$

The proof is finally completed by invoking equations (2), (23) and (26).

## B PROOF OF THEOREM 2

By virtue of the Chernoff bound (1),

$$\begin{aligned}
P(U, \Phi) &= \frac{1}{2} \prod_{m=1}^M \left[ 1 + \frac{\rho^2 \sigma_m^2}{4(1+2\rho)} \right]^{-N} \\
&= \frac{\alpha^{MN}}{2} \left[ \prod_{m=1}^M (\alpha + \sigma_m^2) \right]^{-N}
\end{aligned}$$

where  $\sigma_m$  is the  $m$ -th singular value of  $U - \Phi$ . Let  $U_1$  and  $U_2$  be unitary matrices such that  $U - \Phi = U_1 \text{diag}(\sigma_1, \dots, \sigma_M) U_2^\dagger$ . Then

$$\begin{aligned}
&\det[\alpha I + (U - \Phi)(U - \Phi)^\dagger] \\
&= \det[\alpha I + U_1 \text{diag}(\sigma_1^2, \dots, \sigma_M^2) U_1^\dagger] \\
&= \det[U_1 \text{diag}(\alpha + \sigma_1^2, \dots, \alpha + \sigma_M^2) U_1^\dagger] \\
&= \det(U_1 U_1^\dagger) \det[\text{diag}(\alpha + \sigma_1^2, \dots, \alpha + \sigma_M^2)] \\
&= \prod_{m=1}^M (\alpha + \sigma_m^2).
\end{aligned} \tag{27}$$

It follows that

$$P(U, \Phi) = \frac{\alpha^{MN}}{2 (\det[\alpha I + (U - \Phi)(U - \Phi)^\dagger])^N} \tag{28}$$

from which we obtain

$$\begin{aligned}
P(I, \Phi) &= \frac{\alpha^{MN}}{2 (\det[\alpha I + (I - \Phi)(I - \Phi)^\dagger])^N} \\
&= \frac{\alpha^{MN}}{2 (\det[(\alpha + 2)I - \Phi - \Phi^\dagger])^N}
\end{aligned}$$

by letting  $U = I$ . This proves (3).

Now define

$$\begin{aligned}
\Xi &\stackrel{\text{def}}{=} \log \det[\alpha I + (U - \Phi)(U - \Phi)^\dagger] \\
&= \log \det[(\alpha + 2)I - U\Phi^\dagger - \Phi U^\dagger].
\end{aligned} \tag{29}$$

By the chain rule of differentiation,

$$\begin{aligned}
\frac{\partial P(U, \Phi)}{\partial \nu_{pq}} &= \frac{\partial P(U, \Phi)}{\partial \Xi} \frac{\partial \Xi}{\partial \nu_{pq}} \\
&= \frac{\partial \left[ \frac{\alpha^{MN}}{2} \exp(-N\Xi) \right]}{\partial \Xi} \frac{\partial \Xi}{\partial \nu_{pq}} \\
&= -NP(U, \Phi) \frac{\partial \Xi}{\partial \nu_{pq}}.
\end{aligned} \tag{30}$$

Similarly,

$$\frac{\partial P(U, \Phi)}{\partial \phi_{pq}} = -NP(U, \Phi) \frac{\partial \Xi}{\partial \phi_{pq}}, \tag{31}$$

$$\frac{\partial P(U, \Phi)}{\partial \theta_k} = -NP(U, \Phi) \frac{\partial \Xi}{\partial \theta_k}. \tag{32}$$

Define

$$\Omega \stackrel{\text{def}}{=} (\alpha + 2)I - U\Phi^\dagger - \Phi U^\dagger.$$

By (27) and (29),

$$\det(\Omega) = \prod_{m=1}^M (\alpha + \sigma_m^2) \geq \alpha^M > 0$$

for any  $U$ . Let  $e_j$  be the  $M$ -dimensional unit column vector with a one in the  $j$ -th entry and zeros elsewhere. By the same method of proving (27), we can show that  $\det[(\alpha + 2)I - (U + e_j e_k^\top \delta)\Phi^\dagger - \Phi(U^\dagger + e_k e_j^\top \delta)]$  is a positive real number for any  $\delta \in \mathbb{R}$ . Since  $\det(\Omega)$  is positive and

$$\begin{aligned}
&\det[(\alpha + 2)I - (U + e_j e_k^\top \delta)\Phi^\dagger - \Phi(U^\dagger + e_k e_j^\top \delta)] \\
&= \det(\Omega) \det[I - \Omega^{-1}(e_j e_k^\top \Phi^\dagger + \Phi e_k e_j^\top) \delta],
\end{aligned}$$

we have that  $\det[I - \Omega^{-1}(e_j e_k^\top \Phi^\dagger + \Phi e_k e_j^\top) \delta]$  is also a positive real number for any  $\delta \in \mathbb{R}$ . Therefore,

$$\begin{aligned}
&\log \det[(\alpha + 2)I - (U + e_j e_k^\top \delta)\Phi^\dagger - \Phi(U^\dagger + e_k e_j^\top \delta)] \\
&= \log \det(\Omega) + \log \det[I - \Omega^{-1}(e_j e_k^\top \Phi^\dagger + \Phi e_k e_j^\top) \delta].
\end{aligned}$$

Let  $\Psi = \Omega^{-1}(e_j e_k^\top \Phi^\dagger + \Phi e_k e_j^\top)$ . Then  $\Psi$  is a Hermite matrix, i.e.,  $\Psi^\dagger = \Psi$ . It follows that  $[\Psi]_{kk}$  is real for  $k = 1, \dots, M$ . By the definition of a determinant, we have

$$\det(I - \Psi\delta) = \prod_{k=1}^M (1 - [\Psi]_{kk}\delta) + \delta^2 f(\delta) > 0$$



where  $f(\cdot)$  is a polynomial function of  $\delta \in \mathbb{R}$ . Since  $\det(I - \Psi\delta)$  and  $[\Psi]_{kk}$ ,  $k = 1, \dots, M$  are real numbers, it must be true that  $f(\delta)$  is also a real-valued function of  $\delta \in \mathbb{R}$ . Note that

$$\begin{aligned}\det(I - \Psi\delta) &= 1 - \left( \sum_{k=1}^M [\Psi]_{kk} \right) \delta + O(\delta^2) + \delta^2 f(\delta) \\ &= 1 - \text{tr}(\Psi)\delta + O(\delta^2) \\ &> 0\end{aligned}$$

where  $\text{tr}(\Psi)$  is real and  $O(\delta^2)$  is a real-valued function of  $\delta \in \mathbb{R}$ . Therefore,

$$\log \det(I - \Psi\delta) = -\text{tr}(\Psi)\delta + O(\delta^2). \quad (33)$$

Making use of (33), we have

$$\begin{aligned}& \log \det[(\alpha + 2)I - (U + e_j e_k^\top \delta) \Phi^\dagger - \Phi(U^\dagger + e_k e_j^\top \delta)] \\ &= \log \det(\Omega) - \text{tr}(\Psi\delta) + O(\delta^2) \\ &= \log \det(\Omega) \\ &\quad - \left[ \text{tr} \left( e_j e_k^\top \Phi^\dagger \Omega^{-1} \right) + \text{tr} \left( \Omega^{-1} \Phi e_k e_j^\top \right) \right] \delta + O(\delta^2) \\ &= \log \det(\Omega) \\ &\quad - \left[ \text{tr} \left( e_j e_k^\top (\Omega^{-1} \Phi)^\dagger \right) + \text{tr} \left( \Omega^{-1} \Phi e_k e_j^\top \right) \right] \delta + O(\delta^2) \\ &= \log \det(\Omega) \\ &\quad - \left[ \text{tr} \left( (\Omega^{-1} \Phi e_k e_j^\top)^\dagger \right) + \text{tr} \left( \Omega^{-1} \Phi e_k e_j^\top \right) \right] \delta + O(\delta^2) \\ &= \log \det(\Omega) \\ &\quad - \left[ (\text{tr} \left( \Omega^{-1} \Phi e_k e_j^\top \right))^\dagger + \text{tr} \left( \Omega^{-1} \Phi e_k e_j^\top \right) \right] \delta + O(\delta^2) \\ &= \log \det(\Omega) - 2 \Re(\text{tr} \left( \Omega^{-1} \Phi e_k e_j^\top \right)) \delta + O(\delta^2) \\ &= \log \det(\Omega) - 2 \Re([\Omega^{-1} \Phi]_{jk}) \delta + O(\delta^2) \\ &= \log \det(\Omega) - 2 [\Re(\Omega^{-1} \Phi)]_{jk} \delta + O(\delta^2)\end{aligned}$$

for any  $\delta \in \mathbb{R}$ . Therefore, applying formula

$$\left[ \frac{\partial f(X)}{\partial \Re(X)} \right]_{jk} = \lim_{\delta \rightarrow 0} \frac{f(X + e_j e_k^\top \delta) - f(X)}{\delta}$$

provided in [16] (page 1501), we have

$$\left[ \frac{\partial \Xi}{\partial \Re(U)} \right]_{jk}$$

$$\begin{aligned}
&= \lim_{\delta \rightarrow 0} \frac{\log \det[\Omega - (e_j e_k^\top \Phi^\dagger + \Phi e_k e_j^\top) \delta] - \log \det(\Omega)}{\delta} \\
&= \lim_{\delta \rightarrow 0} \frac{-2[\Re(\Omega^{-1} \Phi)]_{jk} \delta + O(\delta^2)}{\delta} \\
&= -2[\Re(\Omega^{-1} \Phi)]_{jk}.
\end{aligned}$$

Observing that  $U = I$  for  $\Theta = 0$  (i.e., all elements of  $\Theta$  are zeros), we have  $\Omega = (\alpha + 2)I - \Phi - \Phi^\dagger$  and  $\Omega^{-1} \Phi = [(\alpha + 2)I - \Phi - \Phi^\dagger]^{-1} \Phi = \mathcal{Q}$ . Hence,

$$\left. \frac{\partial \Xi}{\partial \Re(U)} \right|_{\Theta=0} = -2\Re(\mathcal{Q}). \quad (34)$$

Similarly,

$$\begin{aligned}
&\log \det[(\alpha + 2)I - (U + e_j e_k^\top \delta i) \Phi^\dagger - \Phi(U^\dagger - e_k e_j^\top \delta i)] \\
&= \log \det[\Omega - (e_j e_k^\top \Phi^\dagger - \Phi e_k e_j^\top) \delta i] \\
&= \log \det(\Omega) + \log \det[I - \Omega^{-1}(e_j e_k^\top \Phi^\dagger - \Phi e_k e_j^\top) \delta i] \\
&= \log \det(\Omega) - \text{tr}[\Omega^{-1}(e_j e_k^\top \Phi^\dagger - \Phi e_k e_j^\top) \delta i] + O(\delta^2) \\
&= \log \det(\Omega) \\
&\quad - \left[ \text{tr} \left( e_j e_k^\top \Phi^\dagger \Omega^{-1} \right) - \text{tr} \left( \Omega^{-1} \Phi e_k e_j^\top \right) \right] \delta i + O(\delta^2) \\
&= \log \det(\Omega) \\
&\quad - \left[ \text{tr} \left( e_j e_k^\top (\Omega^{-1} \Phi)^\dagger \right) - \text{tr} \left( \Omega^{-1} \Phi e_k e_j^\top \right) \right] \delta i + O(\delta^2) \\
&= \log \det(\Omega) \\
&\quad - \left[ \text{tr} \left( (\Omega^{-1} \Phi e_k e_j^\top)^\dagger \right) - \text{tr} \left( \Omega^{-1} \Phi e_k e_j^\top \right) \right] \delta i + O(\delta^2) \\
&= \log \det(\Omega) \\
&\quad - \left[ (\text{tr} \left( \Omega^{-1} \Phi e_k e_j^\top \right))^\dagger - \text{tr} \left( \Omega^{-1} \Phi e_k e_j^\top \right) \right] \delta i + O(\delta^2) \\
&= \log \det(\Omega) - 2 \Im(\text{tr} \left( \Omega^{-1} \Phi e_k e_j^\top \right)) \delta + O(\delta^2) \\
&= \log \det(\Omega) - 2 \Im([\Omega^{-1} \Phi]_{jk}) \delta + O(\delta^2) \\
&= \log \det(\Omega) - 2 [\Im(\Omega^{-1} \Phi)]_{jk} \delta + O(\delta^2)
\end{aligned}$$

for any  $\delta \in \mathbb{R}$ . Therefore,

$$\left[ \frac{\partial \Xi}{\partial \Im(U)} \right]_{jk}$$

$$\begin{aligned}
&= \lim_{\delta \rightarrow 0} \frac{\log \det[\Omega - (e_j e_k^\top \Phi^\dagger - \Phi e_k e_j^\top) \delta i] - \log \det(\Omega)}{\delta} \\
&= \lim_{\delta \rightarrow 0} \frac{-2[\Im(\Omega^{-1} \Phi)]_{jk} \delta + O(\delta^2)}{\delta} \\
&= -2[\Im(\mathcal{Q})]_{jk}
\end{aligned}$$

for  $\Theta = 0$ , which implies that

$$\left. \frac{\partial \Xi}{\partial \Im(U)} \right|_{\Theta=0} = -2\Im(\mathcal{Q}). \quad (35)$$

We now consider the partial derivatives of  $U$  with respect to the elements of  $\Theta$ . It should be noted that an incorrect formula for computing  $\left. \frac{\partial U}{\partial \phi_{pq}} \right|_{\Theta=0}$  has been reported in [1] (see equation (13), page 2625). In the sequel, we shall prove that

$$\left. \frac{\partial U}{\partial \phi_{pq}} \right|_{\Theta=0} = e_q e_p^\top - e_p e_q^\top, \quad (36)$$

which is clearly different from equation (13) of [1]. To that end, we can use the parameterization of unitary matrix  $U(\Theta)$  to verify that

$$\left. \frac{\partial U}{\partial \phi_{pq}} \right|_{\Theta=0} = \left. \frac{\partial \tilde{U}}{\partial \phi_{pq}} \right|_{\phi_{pq}=0}$$

where

$$\tilde{U} = \begin{bmatrix} I_{(p-1) \times (p-1)} & 0_{(p-1) \times (M-p+1)} \\ 0_{(M-p+1) \times (p-1)} & U^{p,q}(\phi_{pq}, 0) \end{bmatrix}$$

with

$$\begin{aligned}
&[U^{p,q}(\phi_{pq}, 0)]_{jk} \\
&= \begin{cases} 1, & \text{if } j = k \text{ and } j \notin \{1, q-p+1\} \\ \cos(\phi_{pq}), & \text{if } j = k \text{ and } j \in \{1, q-p+1\} \\ -\sin(\phi_{pq}), & \text{if } j = 1 \text{ and } k = q-p+1 \\ \sin(\phi_{pq}), & \text{if } k = 1 \text{ and } j = q-p+1 \\ 0, & \text{otherwise.} \end{cases}
\end{aligned}$$

Obviously,

$$\begin{aligned}
&\left. \frac{\partial ([U^{p,q}(\phi_{pq}, 0)]_{jk})}{\partial \phi_{pq}} \right|_{\phi_{pq}=0} \\
&= \begin{cases} -1, & \text{if } j = 1 \text{ and } k = q-p+1 \\ 1, & \text{if } k = 1 \text{ and } j = q-p+1 \\ 0, & \text{otherwise.} \end{cases}
\end{aligned}$$

Hence, (36) can be obtained by observing that

$$\begin{aligned} [\tilde{U}]_{pq} &= [U^{p,q}(\phi_{pq}, 0)]_{1,q-p+1}, \\ [\tilde{U}]_{qp} &= [U^{p,q}(\phi_{pq}, 0)]_{q-p+1,1}. \end{aligned}$$

To compute other partial derivatives of  $U(\Theta)$  at  $\Theta = 0$ , we quote equations (14) and (15) of [1] as follows:

$$\left. \frac{\partial U}{\partial \nu_{pq}} \right|_{\Theta=0} = 0, \quad \left. \frac{\partial U}{\partial \theta_k} \right|_{\Theta=0} = ie_k e_k^\top. \quad (37)$$

By virtue of (36) and (37),

$$\left. \frac{\partial \Re(U)}{\partial \phi_{pq}} \right|_{\Theta=0} = e_q e_p^\top - e_p e_q^\top, \quad \left. \frac{\partial \Im(U)}{\partial \phi_{pq}} \right|_{\Theta=0} = 0, \quad (38)$$

$$\left. \frac{\partial \Re(U)}{\partial \nu_{pq}} \right|_{\Theta=0} = 0, \quad \left. \frac{\partial \Im(U)}{\partial \nu_{pq}} \right|_{\Theta=0} = 0, \quad (39)$$

$$\left. \frac{\partial \Re(U)}{\partial \theta_k} \right|_{\Theta=0} = 0, \quad \left. \frac{\partial \Im(U)}{\partial \theta_k} \right|_{\Theta=0} = e_k e_k^\top. \quad (40)$$

We now define inner product  $\langle \cdot, \cdot \rangle$  by

$$\langle X, Y \rangle \stackrel{\text{def}}{=} \sum_{j,k} [X]_{jk} [Y]_{jk}.$$

Then by the chain rule of differentiation and equations (34), (38), we have

$$\begin{aligned} \frac{\partial \Xi}{\partial \phi_{pq}} &= \left\langle \frac{\partial \Xi}{\partial \Re(U)}, \frac{\partial \Re(U)}{\partial \phi_{pq}} \right\rangle + \left\langle \frac{\partial \Xi}{\partial \Im(U)}, \frac{\partial \Im(U)}{\partial \phi_{pq}} \right\rangle \\ &= \left\langle \frac{\partial \Xi}{\partial \Re(U)}, \frac{\partial \Re(U)}{\partial \phi_{pq}} \right\rangle \\ &= \langle -2 \Re(\mathcal{Q}), e_q e_p^\top - e_p e_q^\top \rangle \\ &= -2 \Re(\langle \mathcal{Q}, e_q e_p^\top - e_p e_q^\top \rangle) \\ &= -2 \Re([\mathcal{Q}]_{qp} - [\mathcal{Q}]_{pq}). \end{aligned}$$

Invoking (31) yields

$$\begin{aligned} \frac{\partial P(U, \Phi)}{\partial \phi_{pq}} &= -NP(U, \Phi) \frac{\partial \Xi}{\partial \phi_{pq}} \\ &= 2NP(U, \Phi) \Re([\mathcal{Q}]_{qp} - [\mathcal{Q}]_{pq}) \end{aligned}$$

and hence proves (4).

By the chain rule of differentiation and (39),

$$\frac{\partial \Xi}{\partial \nu_{pq}} = \left\langle \frac{\partial \Xi}{\partial \Re(U)}, \frac{\partial \Re(U)}{\partial \nu_{pq}} \right\rangle + \left\langle \frac{\partial \Xi}{\partial \Im(U)}, \frac{\partial \Im(U)}{\partial \nu_{pq}} \right\rangle = 0. \quad (41)$$

Combing (30) and (41) leads to

$$\frac{\partial P(U, \Phi)}{\partial \nu_{pq}} = 0$$

and thus completes the proof of (6).

By the chain rule of differentiation and (40),

$$\begin{aligned} \frac{\partial \Xi}{\partial \theta_k} &= \left\langle \frac{\partial \Xi}{\partial \Re(U)}, \frac{\partial \Re(U)}{\partial \theta_k} \right\rangle + \left\langle \frac{\partial \Xi}{\partial \Im(U)}, \frac{\partial \Im(U)}{\partial \theta_k} \right\rangle \\ &= \left\langle \frac{\partial \Xi}{\partial \Im(U)}, \frac{\partial \Im(U)}{\partial \theta_k} \right\rangle \\ &= \langle -2 \Im(\mathcal{Q}), e_k e_k^\top \rangle \\ &= -2 \Im(\langle \mathcal{Q}, e_k e_k^\top \rangle) \\ &= -2 \Im([\mathcal{Q}]_{kk}). \end{aligned}$$

Hence by (32), we have

$$\begin{aligned} \frac{\partial P(U, \Phi)}{\partial \theta_k} &= -NP(U, \Phi) \frac{\partial \Xi}{\partial \theta_k} \\ &= 2NP(U, \Phi) \Im([\mathcal{Q}]_{kk}) \end{aligned}$$

and completes the proof of (5).

Define

$$\Upsilon \stackrel{\text{def}}{=} \log \det[\alpha I + (\Lambda^\ell - \Phi)(\Lambda^\ell - \Phi)^\dagger].$$

By the same method as computing  $\frac{\partial \Xi}{\partial \Im(U)}$ , we have

$$\frac{\partial \Upsilon}{\partial \Im(\Lambda^\ell)} = -2\Im(\mathcal{Q}). \quad (42)$$

Observing that  $[\Lambda^\ell]_{jk}$  depends on  $\lambda_m$  only if  $j = k = m$  and that

$$\left. \frac{\partial \Re([\Lambda^\ell]_{mm})}{\partial \lambda_m} \right|_{\Lambda=I} = \left. \frac{\partial \cos(\frac{2\pi\ell\lambda_m}{L})}{\partial \lambda_m} \right|_{\Lambda=I} = 0$$

$$\left. \frac{\partial \Im([\Lambda^\ell]_{mm})}{\partial \lambda_m} \right|_{\Lambda=I} = \left. \frac{\partial \sin(\frac{2\pi\ell\lambda_m}{L})}{\partial \lambda_m} \right|_{\Lambda=I} = \frac{2\pi\ell}{L},$$

we have

$$\left. \frac{\partial \Re(\Lambda^\ell)}{\partial \lambda_m} \right|_{\Lambda=I} = 0, \quad \left. \frac{\partial \Im(\Lambda^\ell)}{\partial \lambda_m} \right|_{\Lambda=I} = \frac{2\pi\ell}{L} e_m e_m^\top. \quad (43)$$

By the chain rule of differentiation and equations (42), (43), we have

$$\begin{aligned} \frac{\partial \Upsilon}{\partial \lambda_m} &= \left\langle \frac{\partial \Upsilon}{\partial \Re(\Lambda^\ell)}, \frac{\partial \Re(\Lambda^\ell)}{\partial \lambda_m} \right\rangle + \left\langle \frac{\partial \Upsilon}{\partial \Im(\Lambda^\ell)}, \frac{\partial \Im(\Lambda^\ell)}{\partial \lambda_m} \right\rangle \\ &= \left\langle \frac{\partial \Upsilon}{\partial \Im(\Lambda^\ell)}, \frac{\partial \Im(\Lambda^\ell)}{\partial \lambda_m} \right\rangle \\ &= \left\langle -2\Im(\mathcal{Q}), \frac{2\pi\ell}{L} e_m e_m^\top \right\rangle \\ &= -\frac{4\pi\ell}{L} \Im([\mathcal{Q}]_{mm}). \end{aligned}$$

It follows that

$$\begin{aligned} \frac{\partial P(\Lambda^\ell, \Phi)}{\partial \lambda_m} &= -NP(\Lambda^\ell, \Phi) \frac{\partial \Upsilon}{\partial \lambda_m} \\ &= \frac{4\pi\ell NP(\Lambda^\ell, \Phi)}{L} \Im([\mathcal{Q}]_{mm}) \end{aligned}$$

and (7) is true.

## C PROOF OF THEOREM 3

First we need to prove some preliminary results.

**Lemma 1** *For any  $\ell \in \{0, 1, \dots, L-1\}$ , there exists  $y \in \mathcal{S}$  such that*

$$\sum_{m=1}^M [(C_m \lambda_m \ell - C_m \varphi_m) \bmod^* C_m L]^2 = \|y G - \xi\|^2. \quad (44)$$

**Proof.** Given  $\ell \in \{0, 1, \dots, L-1\}$ , define

$$y_1 = \ell + \left\lceil \frac{\varphi_1 - \ell}{L} - \frac{1}{2} \right\rceil L. \quad (45)$$

We claim that

$$-\frac{L}{2} + \varphi_1 \leq y_1 < \frac{L}{2} + \varphi_1. \quad (46)$$

To prove (46), one can make use of the observation that

$$0 \leq \lceil x \rceil - x < 1 \quad \forall x \in \mathbb{R}$$

and verify that inequality

$$0 \leq \left\lceil \frac{\varphi_1 - \ell}{L} - \frac{1}{2} \right\rceil - \left( \frac{\varphi_1 - \ell}{L} - \frac{1}{2} \right) < 1$$

is equivalent to

$$-\frac{L}{2} + \varphi_1 \leq \ell + \left\lceil \frac{\varphi_1 - \ell}{L} - \frac{1}{2} \right\rceil L < \frac{L}{2} + \varphi_1.$$

The truth of (46) allows one to choose  $y = [y_1, \dots, y_M] \in \mathcal{S}$  such that the first entry of  $y$  is  $y_1$ . Let

$$w = [w_1, \dots, w_M] = y G - \xi. \quad (47)$$

Obviously, to show (44), it suffices to show

$$(C_m \lambda_m \ell - C_m \varphi_m) \bmod^* C_m L = w_m, \quad m = 1, \dots, M$$

where  $\lambda_1 = 1$ . By the definitions of  $G$  and  $\xi$ , we can rewrite (47) as

$$w_1 = C_1 y_1 - C_1 \varphi_1,$$

$$w_m = C_m \lambda_m y_1 + C_m L y_m - C_m \varphi_m \quad \text{for } m = 2, \dots, M.$$

Hence, to show (44), it suffices to show

$$(C_1 \ell - C_1 \varphi_1) \bmod^* C_1 L = C_1 y_1 - C_1 \varphi_1 \quad (48)$$

and, for  $m = 2, \dots, M$ ,

$$\begin{aligned} & (C_m \lambda_m \ell - C_m \varphi_m) \bmod^* C_m L \\ &= C_m \lambda_m y_1 + C_m L y_m - C_m \varphi_m. \end{aligned} \quad (49)$$

Note that, for any  $\ell$ , there exists a unique integer  $z_1$  such that

$$(C_1 \ell - C_1 \varphi_1) \bmod^* C_1 L = C_1 \ell - C_1 \varphi_1 + z_1 C_1 L.$$

Therefore, to show (48), it suffices to show

$$C_1 y_1 - C_1 \varphi_1 = C_1 \ell - C_1 \varphi_1 + z_1 C_1 L,$$

or equivalently,

$$z_1 = \frac{y_1 - \ell}{L}. \quad (50)$$

By the definition of the symmetric modulus operator  $\bmod^*$ , integer  $z_1$  guarantees

$$-\frac{C_1 L}{2} \leq C_1 \ell - C_1 \varphi_1 + z_1 C_1 L < \frac{C_1 L}{2},$$

or equivalently,

$$-1 < \left( \frac{\varphi_1 - \ell}{L} - \frac{1}{2} \right) - z_1 \leq 0,$$

which implies

$$\left\lceil \frac{\varphi_1 - \ell}{L} - \frac{1}{2} - z_1 \right\rceil = 0.$$

Since  $z_1$  is an integer, we have

$$\begin{aligned} z_1 &= \left\lceil \frac{\varphi_1 - \ell}{L} - \frac{1}{2} \right\rceil \\ &= \frac{y_1 - \ell}{L} \end{aligned}$$

where the second equality follows from (45). So equation (48) is proven by invoking (50).

In light of the fact that, for any given  $\ell \in \{0, 1, \dots, L-1\}$  and for any  $m \in \{2, \dots, M\}$ , there exists a unique integer  $z_m$  such that

$$(C_m \lambda_m \ell - C_m \varphi_m) \bmod^* C_m L = C_m (\lambda_m \ell - \varphi_m + z_m L),$$

to show (49) it suffices to prove that

$$C_m (\lambda_m \ell - \varphi_m + z_m L) = C_m \lambda_m y_1 + C_m L y_m - C_m \varphi_m$$

for  $m = 2, \dots, M$ , or equivalently,

$$y_m = z_m - \frac{y_1 - \ell}{L} \lambda_m, \quad m = 2, \dots, M. \quad (51)$$



By the definition of the symmetric modulus operator  $\text{mod}^*$ , integer  $z_m$  guarantees

$$-\frac{C_m L}{2} \leq C_m \lambda_m \ell - C_m \varphi_m + z_m C_m L < \frac{C_m L}{2}$$

which can be rewritten as

$$-\frac{C_m L}{2} \leq C_m [\lambda_m (\ell + z_1 L) - \varphi_m + (z_m - z_1 \lambda_m) L] < \frac{C_m L}{2},$$

i.e.,

$$-1 < \left( \frac{\varphi_m}{L} - \left( \frac{\ell + z_1 L}{L} - z_1 \right) \lambda_m - \frac{1}{2} \right) - z_m \leq 0$$

for  $m = 2, \dots, M$ . Therefore,

$$\left\lceil \left( \frac{\varphi_m}{L} - \left( \frac{\ell + z_1 L}{L} - z_1 \right) \lambda_m - \frac{1}{2} \right) - z_m \right\rceil = 0$$

for  $m = 2, \dots, M$ . Since  $z_m$  is an integer, we have

$$\begin{aligned} z_m &= \left\lceil \frac{\varphi_m}{L} - \left( \frac{\ell + z_1 L}{L} - z_1 \right) \lambda_m - \frac{1}{2} \right\rceil \\ &= \left\lceil \frac{\varphi_m}{L} - \left( \frac{y_1}{L} - z_1 \right) \lambda_m - \frac{1}{2} \right\rceil \end{aligned} \quad (52)$$

for  $m = 2, \dots, M$ . Here (52) is due to (50). By the definition of  $\mathcal{S}$ ,

$$y_m = \left\lceil \frac{\varphi_m}{L} - \left( \frac{y_1}{L} - \left\lfloor \frac{y_1}{L} \right\rfloor \right) \lambda_m - \frac{1}{2} \right\rceil - \left\lfloor \frac{y_1}{L} \right\rfloor \lambda_m. \quad (53)$$

By virtue of (50) and the fact that  $0 \leq \ell < L$ , we have

$$0 \leq \frac{y_1}{L} - z_1 < 1,$$

which leads to

$$\left\lfloor \frac{y_1}{L} - z_1 \right\rfloor = 0$$

and consequently

$$z_1 = \left\lfloor \frac{y_1}{L} \right\rfloor. \quad (54)$$

Combining (50), (52), (53) and (54) yields

$$\begin{aligned} y_m &= \left\lceil \frac{\varphi_m}{L} - \left( \frac{y_1}{L} - z_1 \right) \lambda_m - \frac{1}{2} \right\rceil - z_1 \lambda_m \\ &= z_m - z_1 \lambda_m \\ &= z_m - \frac{y_1 - \ell}{L} \lambda_m \end{aligned}$$

for  $m = 2, \dots, M$ . This proves (51). It follows that (49) is true and the lemma is thus proven.  $\square$

**Lemma 2** *Let  $y = [y_1, \dots, y_M] \in \mathcal{S}$ . If  $\tilde{\ell} = y_1 - \lfloor \frac{y_1}{L} \rfloor L$ , then  $0 \leq \tilde{\ell} < L$  and*

$$\sum_{m=1}^M [(C_m \lambda_m \tilde{\ell} - C_m \varphi_m) \bmod^* C_m L]^2 = \|y G - \xi\|^2. \quad (55)$$

**Proof.** By the definition of  $\tilde{\ell}$ ,

$$\frac{\tilde{\ell}}{L} = \frac{y_1}{L} - \left\lfloor \frac{y_1}{L} \right\rfloor \in [0, 1).$$

Hence,  $0 \leq \tilde{\ell} < L$ . Clearly, there uniquely exist integers  $\tilde{z}_1, \dots, \tilde{z}_M$  such that, for  $m = 1, \dots, M$ ,

$$(C_m \lambda_m \tilde{\ell} - C_m \varphi_m) \bmod^* C_m L = C_m \lambda_m \tilde{\ell} + \tilde{z}_m C_m L - C_m \varphi_m$$

where  $\lambda_1 = 1$ . Therefore, to prove (55) it suffices to show

$$C_1 \tilde{\ell} + \tilde{z}_1 C_1 L - C_1 \varphi_1 = C_1 y_1 - C_1 \varphi_1 \quad (56)$$

and, for  $m = 2, \dots, M$ ,

$$C_m \lambda_m \tilde{\ell} + \tilde{z}_m C_m L - C_m \varphi_m = C_m \lambda_m y_1 + C_m L y_m - C_m \varphi_m. \quad (57)$$

Equation (56) can be simplified as

$$\tilde{\ell} + \tilde{z}_1 L = y_1, \quad (58)$$

which can be further reduced to

$$\tilde{z}_1 = \left\lfloor \frac{y_1}{L} \right\rfloor \quad (59)$$

by invoking the definition of  $\tilde{\ell}$ . By the definition of the symmetric modulus operator  $\bmod^*$ ,

$$-\frac{C_1 L}{2} \leq C_1 \tilde{\ell} + \tilde{z}_1 C_1 L - C_1 \varphi_1 < \frac{C_1 L}{2},$$

which can be rewritten as

$$-1 < \left( \frac{\varphi_1 - \tilde{\ell}}{L} - \frac{1}{2} \right) - \tilde{z}_1 \leq 0,$$

or equivalently,

$$\left\lceil \left( \frac{\varphi_1 - \tilde{\ell}}{L} - \frac{1}{2} \right) - \tilde{z}_1 \right\rceil = 0. \quad (60)$$

By virtue of (60) and the definition of  $\tilde{\ell}$ ,

$$\begin{aligned} \tilde{z}_1 &= \left\lceil \frac{\varphi_1 - \tilde{\ell}}{L} - \frac{1}{2} \right\rceil \\ &= \left\lceil \frac{\varphi_1 - (y_1 - \lfloor \frac{y_1}{L} \rfloor L)}{L} - \frac{1}{2} \right\rceil \\ &= \left\lceil \frac{\varphi_1}{L} - \frac{y_1}{L} - \frac{1}{2} \right\rceil + \left\lfloor \frac{y_1}{L} \right\rfloor. \end{aligned} \quad (61)$$

Remember that  $y_1$  is restricted by condition

$$-\frac{L}{2} + \varphi_1 \leq y_1 < \frac{L}{2} + \varphi_1,$$

or equivalently

$$-1 < \frac{\varphi_1}{L} - \frac{y_1}{L} - \frac{1}{2} \leq 0$$

which implies

$$\left\lceil \frac{\varphi_1}{L} - \frac{y_1}{L} - \frac{1}{2} \right\rceil = 0. \quad (62)$$

Combining (61) with (62) yields (59) and consequently proves (56).

We now turn our attention to the proof of (57). By the definition of  $\tilde{\ell}$ , (57) can be rewritten as

$$\lambda_m \left( y_1 - \left\lfloor \frac{y_1}{L} \right\rfloor L \right) + \tilde{z}_m L = y_1 \lambda_m + y_m L,$$

which can be further simplified as

$$\tilde{z}_m = y_m + \left\lfloor \frac{y_1}{L} \right\rfloor \lambda_m. \quad (63)$$

By the definition of the symmetric modulus operator  $\text{mod}^*$ , we have

$$-\frac{C_m L}{2} \leq C_m \lambda_m \tilde{\ell} + \tilde{z}_m C_m L - C_m \varphi_m < \frac{C_m L}{2},$$

which can be rewritten as

$$-\frac{C_m L}{2} \leq C_m [\lambda_m (\tilde{\ell} + \tilde{z}_1 L) + (\tilde{z}_m - \tilde{z}_1 \lambda_m) L - \varphi_m] < \frac{C_m L}{2},$$

or equivalently,

$$-1 < \frac{\varphi_m}{L} - \left( \frac{\tilde{\ell} + \tilde{z}_1 L}{L} - \tilde{z}_1 \right) \lambda_m - \frac{1}{2} - \tilde{z}_m \leq 0.$$

Hence,

$$\left\lceil \frac{\varphi_m}{L} - \left( \frac{\tilde{\ell} + \tilde{z}_1 L}{L} - \tilde{z}_1 \right) \lambda_m - \frac{1}{2} - \tilde{z}_m \right\rceil = 0.$$

Since  $\tilde{z}_m$  is an integer, it can be determined that

$$\tilde{z}_m = \left\lceil \frac{\varphi_m}{L} - \left( \frac{\tilde{\ell} + \tilde{z}_1 L}{L} - \tilde{z}_1 \right) \lambda_m - \frac{1}{2} \right\rceil. \quad (64)$$

Note that (58) is true since (59) has been established. Using (58), (59) and (64), we obtain

$$\tilde{z}_m = \left\lceil \frac{\varphi_m}{L} - \left( \frac{y_1}{L} - \left\lfloor \frac{y_1}{L} \right\rfloor \right) \lambda_m - \frac{1}{2} \right\rceil. \quad (65)$$

Invoking (53) and (65) leads to (63). This proves (57) and the proof of the lemma is thus completed.  $\square$

We are now in position to prove Theorem 3. By Lemma 1, we have

$$\min_{\ell} \sum_{m=1}^M [(C_m \lambda_m \ell - C_m \varphi_m) \text{mod}^* C_m L]^2 \geq \min_{y \in \mathcal{S}} \|yG - \xi\|^2.$$

On the other hand, by Lemma 2, we have

$$\min_{\ell} \sum_{m=1}^M [(C_m \lambda_m \ell - C_m \varphi_m) \text{mod}^* C_m L]^2 \leq \min_{y \in \mathcal{S}} \|yG - \xi\|^2.$$

Therefore,

$$\min_{\ell} \sum_{m=1}^M [(C_m \lambda_m \ell - C_m \varphi_m) \bmod^* C_m L]^2 = \min_{y \in \mathcal{S}} \|yG - \xi\|^2.$$

Since  $\widehat{y}_1$  is the first entry of

$$\widehat{y} = [\widehat{y}_1, \dots, \widehat{y}_M] = \arg \min_{y \in \mathcal{S}} \|yG - \xi\|^2$$

and  $\widehat{\ell}$  is unique, it follows from Lemma 2 that

$$\begin{aligned} \widehat{\ell} &= \arg \min_{\ell} \sum_{m=1}^M [(C_m \lambda_m \ell - C_m \varphi_m) \bmod^* C_m L]^2 \\ &= \widehat{y}_1 - \left\lfloor \frac{\widehat{y}_1}{L} \right\rfloor L. \end{aligned}$$

The proof of Theorem 3 is thus completed.

## D PROOF OF THEOREM 4

By the definitions of  $G$  and  $\xi$ ,

$$\|yG - \xi\|^2 = [C_1(y_1 - \varphi_1)]^2 + \sum_{m=2}^M [C_m(\lambda_m y_1 + L y_m - \varphi_m)]^2.$$

Note that

$$[C_1(y_1 - \varphi_1)]^2 + \sum_{m=2}^M [C_m(\lambda_m y_1 + L y_m - \varphi_m)]^2 < \gamma^2$$

if and only if

$$[C_1(y_1 - \varphi_1)]^2 < \gamma^2, \tag{66}$$

$$\begin{aligned} [C_1(y_1 - \varphi_1)]^2 + \sum_{p=2}^m [C_p(\lambda_p y_1 + L y_p - \varphi_p)]^2 &< \gamma^2 \\ \text{for } m &= 2, \dots, M. \end{aligned} \tag{67}$$

Inequalities (66) and (67) can be shown to be equivalent to (13) and (14) by invoking the definitions of  $\mathcal{S}$  and  $\mu_m$ .

## E DATA OF UNITARY SPACE-TIME CODES

Constellation for  $M = N = b = 2$ ,  $L = 1024$ ,  $R = 6$

For  $q = 0, 1, 2, 3$ ,

$$\Lambda_q = \text{diag} \left( \exp \left( \frac{2\pi i}{L} [1 \ 376] \right) \right),$$

$$A_q = B_0 = I_{2 \times 2}.$$

$$B_1 = \begin{bmatrix} 0.5192 + 0.1730i & 0.7689 + 0.3305i \\ 0.3249 + 0.7713i & -0.1692 - 0.5205i \end{bmatrix},$$

$$B_2 = \begin{bmatrix} 0.4772 - 0.3219i & 0.0907 + 0.8127i \\ -0.1774 + 0.7983i & 0.4398 + 0.3713i \end{bmatrix},$$

$$B_3 = \begin{bmatrix} -0.4458 + 0.3772i & 0.7645 + 0.2729i \\ -0.6303 + 0.5115i & -0.5459 - 0.2075i \end{bmatrix}.$$

See Figure 3 for the corresponding performance simulation results.

Constellation for  $M = N = 2$ ,  $b = 3$ ,  $L = 512$ ,  $R = 6$

For  $q = 0, 1, \dots, 7$ ,

$$\Lambda_q = \text{diag} \left( \exp \left( \frac{2\pi i}{L} [1 \ 188] \right) \right),$$

$$A_q = B_0 = I_{2 \times 2}.$$

$$B_1 = \begin{bmatrix} 0.3408 + 0.6630i & -0.1400 - 0.6517i \\ -0.2401 + 0.6218i & -0.4402 + 0.6016i \end{bmatrix},$$

$$B_2 = \begin{bmatrix} 0.4230 + 0.2881i & -0.7279 + 0.4563i \\ 0.8585 + 0.0319i & 0.2226 - 0.4609i \end{bmatrix},$$

$$\begin{aligned}
B_3 &= \begin{bmatrix} 0.3663 + 0.1357i & 0.8257 + 0.4069i \\ -0.5379 - 0.7470i & 0.1944 + 0.3388i \end{bmatrix}, \\
B_4 &= \begin{bmatrix} 0.7428 + 0.1845i & 0.4221 - 0.4858i \\ -0.6130 - 0.1962i & 0.5391 - 0.5433i \end{bmatrix}, \\
B_5 &= \begin{bmatrix} 0.2656 - 0.1927i & -0.9238 + 0.1975i \\ 0.3009 + 0.8954i & -0.0304 + 0.3267i \end{bmatrix}, \\
B_6 &= \begin{bmatrix} 0.0816 + 0.8219i & -0.4396 - 0.3530i \\ 0.3081 + 0.4722i & 0.8099 + 0.1620i \end{bmatrix}, \\
B_7 &= \begin{bmatrix} -0.0442 - 0.7407i & 0.5677 - 0.3564i \\ -0.5320 - 0.4079i & -0.1131 + 0.7334i \end{bmatrix}.
\end{aligned}$$

See Figure 3 for the corresponding performance simulation results.

Constellation for  $M = N = 2$ ,  $b = 4$ ,  $L = 256$ ,  $R = 6$

For  $q = 0, 1, \dots, 15$ ,

$$\begin{aligned}
\Lambda_q &= \text{diag} \left( \exp \left( \frac{2\pi i}{L} [1 \ 75.7044] \right) \right), \\
A_q &= B_0 = I_{2 \times 2}.
\end{aligned}$$

$$\begin{aligned}
B_1 &= \begin{bmatrix} 0.3912 - 0.8587i & 0.1204 - 0.3083i \\ 0.2004 + 0.2635i & -0.6117 - 0.7185i \end{bmatrix}, \\
B_2 &= \begin{bmatrix} -0.1412 + 0.1279i & 0.1820 + 0.9647i \\ 0.7979 - 0.5718i & 0.0138 + 0.1900i \end{bmatrix}, \\
B_3 &= \begin{bmatrix} 0.4099 + 0.6855i & 0.5015 - 0.3325i \\ -0.5988 - 0.0590i & 0.0412 - 0.7976i \end{bmatrix}, \\
B_4 &= \begin{bmatrix} 0.2787 + 0.3877i & -0.6636 - 0.5759i \\ 0.8235 + 0.3064i & 0.4739 + 0.0588i \end{bmatrix}, \\
B_5 &= \begin{bmatrix} -0.7060 + 0.4436i & -0.3287 - 0.4435i \\ 0.5057 - 0.2215i & -0.3922 - 0.7358i \end{bmatrix}, \\
B_6 &= \begin{bmatrix} 0.5580 + 0.0014i & -0.0343 - 0.8292i \\ 0.7699 - 0.3096i & 0.2307 + 0.5080i \end{bmatrix},
\end{aligned}$$

$$\begin{aligned}
B_7 &= \begin{bmatrix} -0.2027 + 0.7271i & 0.2565 + 0.6037i \\ -0.6504 - 0.0851i & 0.6461 - 0.3904i \end{bmatrix}, \\
B_8 &= \begin{bmatrix} -0.2526 + 0.2877i & -0.2635 - 0.8854i \\ 0.3749 - 0.8443i & -0.2137 - 0.3177i \end{bmatrix}, \\
B_9 &= \begin{bmatrix} -0.8735 + 0.1501i & 0.2828 - 0.3666i \\ 0.4626 + 0.0203i & 0.6780 - 0.5708i \end{bmatrix}, \\
B_{10} &= \begin{bmatrix} -0.8175 - 0.4231i & -0.0195 + 0.3903i \\ 0.2927 - 0.2589i & 0.8409 + 0.3744i \end{bmatrix}, \\
B_{11} &= \begin{bmatrix} -0.5913 - 0.4068i & -0.4858 - 0.4989i \\ -0.6674 - 0.1988i & 0.6347 + 0.3350i \end{bmatrix}, \\
B_{12} &= \begin{bmatrix} 0.3609 + 0.1064i & 0.7232 + 0.5792i \\ 0.9174 + 0.1295i & -0.3248 - 0.1899i \end{bmatrix}, \\
B_{13} &= \begin{bmatrix} -0.1464 - 0.8164i & -0.0346 + 0.5575i \\ -0.4076 + 0.3819i & -0.7225 + 0.4075i \end{bmatrix}, \\
B_{14} &= \begin{bmatrix} -0.0575 + 0.6282i & -0.6849 + 0.3647i \\ 0.3285 + 0.7030i & 0.3313 - 0.5368i \end{bmatrix}, \\
B_{15} &= \begin{bmatrix} 0.4912 + 0.3594i & -0.0364 - 0.7926i \\ 0.3456 + 0.7142i & -0.2606 + 0.5500i \end{bmatrix}.
\end{aligned}$$

See Figure 3 for the corresponding performance simulation results.

Constellation for  $M = 3$ ,  $N = 1$ ,  $b = 4$ ,  $L = 256$ ,  $R = 4$  (see Figure 4)

---

$$\Lambda_q = \text{diag} \left( \exp \left( \frac{2\pi i}{L} [1 \ 33.7365 \ 58.5425] \right) \right), \quad A_q = B_0 = I_{3 \times 3}, \quad q = 0, 1, \dots, 15.$$

$$\begin{aligned}
B_1 &= \begin{bmatrix} 0.7602 + 0.1419i & -0.3318 - 0.3072i & 0.2330 + 0.3785i \\ -0.0629 - 0.2186i & -0.1319 - 0.8171i & -0.1144 - 0.5002i \\ 0.2379 + 0.5419i & -0.1797 + 0.2798i & 0.1680 - 0.7148i \end{bmatrix}, \\
B_2 &= \begin{bmatrix} -0.1626 + 0.2936i & -0.2426 - 0.5136i & 0.2881 + 0.6941i \\ -0.2142 + 0.8692i & 0.3728 + 0.1549i & 0.0315 - 0.1861i \\ 0.0007 + 0.2931i & -0.6903 + 0.1948i & -0.6309 + 0.0408i \end{bmatrix}, \\
B_3 &= \begin{bmatrix} -0.3998 + 0.6106i & -0.6134 + 0.2661i & 0.0346 - 0.1385i \\ 0.0378 - 0.1693i & -0.2662 - 0.0232i & -0.9470 + 0.0426i \\ 0.5614 - 0.3495i & -0.6931 + 0.0357i & 0.2809 + 0.0468i \end{bmatrix},
\end{aligned}$$



$$\begin{aligned}
B_4 &= \begin{bmatrix} -0.0076 + 0.1851i & 0.2440 - 0.9364i & -0.1497 - 0.0827i \\ 0.2850 - 0.6178i & -0.0305 - 0.0343i & -0.2300 - 0.6943i \\ 0.2799 - 0.6515i & 0.1769 - 0.1737i & 0.2094 + 0.6260i \end{bmatrix}, \\
B_5 &= \begin{bmatrix} -0.0148 + 0.0411i & 0.5416 - 0.0737i & 0.1535 - 0.8221i \\ 0.2580 - 0.1099i & -0.5039 + 0.6258i & -0.2723 - 0.4490i \\ 0.9480 - 0.1438i & 0.2031 - 0.1203i & 0.0806 + 0.1354i \end{bmatrix}, \\
B_6 &= \begin{bmatrix} 0.6097 + 0.1555i & 0.0383 + 0.3146i & -0.5723 + 0.4197i \\ 0.4287 - 0.0217i & -0.3612 + 0.5689i & 0.5271 - 0.2896i \\ 0.6463 + 0.0458i & 0.1922 - 0.6392i & 0.1045 - 0.3518i \end{bmatrix}, \\
B_7 &= \begin{bmatrix} -0.0706 - 0.1161i & 0.1815 + 0.0717i & 0.8110 + 0.5345i \\ -0.4231 + 0.4203i & 0.7746 + 0.1128i & -0.1761 + 0.0243i \\ 0.3546 - 0.7072i & 0.5745 - 0.1383i & -0.1161 - 0.1071i \end{bmatrix}, \\
B_8 &= \begin{bmatrix} 0.2373 + 0.6258i & 0.6446 - 0.3587i & -0.0844 - 0.0277i \\ -0.0899 - 0.5506i & 0.5100 - 0.0798i & -0.2909 + 0.5811i \\ -0.2728 + 0.4080i & -0.4228 - 0.1028i & -0.6564 + 0.3728i \end{bmatrix}, \\
B_9 &= \begin{bmatrix} 0.2476 + 0.3168i & -0.3066 - 0.6917i & -0.3383 - 0.3891i \\ 0.4145 - 0.3141i & 0.6045 - 0.2448i & -0.4322 + 0.3426i \\ 0.5080 - 0.5566i & -0.0430 + 0.0190i & 0.4250 - 0.4993i \end{bmatrix}, \\
B_{10} &= \begin{bmatrix} -0.4832 - 0.0060i & -0.4217 + 0.6427i & 0.4049 + 0.1075i \\ 0.2751 - 0.0456i & 0.2416 - 0.2205i & 0.8445 + 0.3195i \\ -0.1269 - 0.8201i & 0.4829 + 0.2623i & -0.0876 + 0.0397i \end{bmatrix}, \\
B_{11} &= \begin{bmatrix} 0.3610 + 0.4449i & 0.2997 + 0.4246i & 0.5036 + 0.3846i \\ -0.2332 + 0.7337i & 0.1987 - 0.0629i & -0.0908 - 0.5963i \\ -0.2399 + 0.1465i & 0.4152 - 0.7170i & 0.0309 + 0.4833i \end{bmatrix}, \\
B_{12} &= \begin{bmatrix} -0.0346 - 0.0609i & -0.6038 + 0.5991i & -0.4153 - 0.3148i \\ 0.0784 + 0.8724i & -0.1136 + 0.2598i & 0.3898 + 0.0206i \\ -0.3106 - 0.3625i & 0.1769 + 0.4060i & 0.6732 - 0.3505i \end{bmatrix}, \\
B_{13} &= \begin{bmatrix} -0.0800 - 0.3611i & -0.4150 + 0.0674i & -0.0114 - 0.8284i \\ -0.0151 + 0.0077i & -0.7741 + 0.4418i & 0.1714 + 0.4195i \\ -0.3292 - 0.8687i & 0.1661 - 0.0354i & 0.0598 + 0.3235i \end{bmatrix}, \\
B_{14} &= \begin{bmatrix} -0.0408 + 0.0395i & 0.4025 + 0.4897i & -0.1679 - 0.7529i \\ 0.8274 - 0.2735i & 0.3768 + 0.1065i & 0.2514 + 0.1554i \\ 0.4822 + 0.0702i & -0.5340 - 0.3997i & -0.2202 - 0.5188i \end{bmatrix},
\end{aligned}$$

$$B_{15} = \begin{bmatrix} 0.4458 + 0.2691i & -0.7835 + 0.2801i & -0.1008 - 0.1623i \\ -0.5050 - 0.1408i & -0.4301 + 0.1993i & 0.1919 + 0.6809i \\ 0.1380 - 0.6595i & -0.2277 - 0.1762i & 0.6115 - 0.2987i \end{bmatrix}.$$

Constellation for  $M = 4$ ,  $N = 2$ ,  $b = 2$ ,  $L = 64$ ,  $R = 2$  (see Figure 5)

$$\Lambda_q = \text{diag} \left( \exp \left( \frac{2\pi i}{L} [1 \ 5 \ 17 \ 28] \right) \right), \quad A_q = B_0 = I_{4 \times 4}, \quad q = 0, 1, 2, 3.$$

$$B_1 = \begin{bmatrix} 0.1920 - 0.0840i & -0.2404 - 0.0482i & 0.4479 - 0.5434i & -0.5535 - 0.3061i \\ 0.2506 - 0.2836i & -0.0749 + 0.0316i & -0.3375 + 0.5975i & -0.5664 - 0.2417i \\ 0.5003 + 0.4453i & -0.3540 - 0.4598i & -0.0062 + 0.1056i & -0.0564 + 0.4475i \\ -0.5925 + 0.1151i & -0.7710 - 0.0446i & -0.1140 + 0.0945i & 0.0034 - 0.1313i \end{bmatrix},$$

$$B_2 = \begin{bmatrix} 0.0820 + 0.1057i & -0.3400 - 0.6390i & -0.1703 - 0.4908i & 0.3557 - 0.2486i \\ -0.1493 - 0.5629i & 0.0330 + 0.1103i & -0.0758 + 0.3064i & 0.7293 - 0.1268i \\ 0.4421 + 0.5263i & 0.2416 + 0.4456i & -0.0185 - 0.1039i & 0.4068 - 0.3064i \\ -0.3881 - 0.1410i & 0.3614 + 0.2743i & -0.5385 - 0.5739i & -0.0546 + 0.0366i \end{bmatrix},$$

$$B_3 = \begin{bmatrix} -0.0069 + 0.0651i & 0.5578 + 0.1900i & -0.2629 - 0.3439i & 0.2290 - 0.6392i \\ 0.3545 - 0.6450i & -0.1827 - 0.2323i & 0.4409 - 0.1617i & 0.0128 - 0.3878i \\ -0.1216 + 0.4867i & -0.4988 + 0.1761i & 0.2493 + 0.1850i & -0.1557 - 0.5899i \\ 0.1847 - 0.4103i & -0.1855 + 0.5012i & -0.4816 + 0.5137i & 0.0359 - 0.1220i \end{bmatrix}.$$

Constellation for  $M = 4$ ,  $N = 2$ ,  $b = 4$ ,  $L = 256$ ,  $R = 3$  (see Figure 6)

$$\Lambda_q = \text{diag} \left( \exp \left( \frac{2\pi i}{L} [1 \ 7.9761 \ 68.6816 \ 106.6000] \right) \right), \quad A_q = B_0 = I_{4 \times 4}, \quad q = 0, 1, \dots, 15.$$

$$B_1 = \begin{bmatrix} -0.4860 - 0.2228i & -0.6620 - 0.2202i & -0.1202 - 0.3650i & 0.0005 + 0.2823i \\ 0.2336 - 0.2148i & 0.2808 + 0.1490i & -0.1188 - 0.6816i & 0.5604 + 0.0744i \\ 0.0589 + 0.3944i & -0.0909 + 0.0457i & 0.0637 - 0.5901i & -0.4779 - 0.5000i \\ 0.6445 - 0.1977i & -0.6320 + 0.0494i & -0.0009 + 0.1460i & 0.1766 - 0.3019i \end{bmatrix},$$

$$B_2 = \begin{bmatrix} -0.4407 + 0.1717i & -0.6272 + 0.0508i & 0.4816 + 0.1003i & -0.2527 + 0.2729i \\ -0.8127 - 0.0589i & 0.4192 + 0.0524i & -0.0967 - 0.3587i & -0.0923 - 0.1050i \\ 0.0371 - 0.3235i & -0.3105 - 0.1548i & -0.5337 - 0.4044i & -0.1209 + 0.5573i \\ 0.0490 - 0.0620i & 0.5423 + 0.1058i & 0.1616 + 0.3814i & -0.2759 + 0.6639i \end{bmatrix},$$

$$\begin{aligned}
B_3 &= \begin{bmatrix} -0.2059 - 0.1255i & 0.0653 + 0.2605i & -0.7272 + 0.4766i & -0.0893 - 0.3254i \\ -0.5978 - 0.0094i & 0.0292 - 0.0758i & 0.3068 - 0.0283i & 0.5017 - 0.5378i \\ -0.0554 - 0.1514i & 0.3235 - 0.8910i & -0.2501 - 0.0408i & -0.1019 - 0.0281i \\ -0.1131 - 0.7386i & -0.1193 + 0.0915i & 0.2566 - 0.1384i & -0.5327 - 0.2241i \end{bmatrix}, \\
B_4 &= \begin{bmatrix} 0.2059 - 0.4636i & 0.6366 + 0.2002i & 0.1927 - 0.2336i & -0.3935 - 0.2254i \\ 0.2157 - 0.2034i & 0.0779 + 0.1506i & -0.0209 + 0.7166i & -0.2077 + 0.5712i \\ -0.1516 - 0.5863i & -0.3966 - 0.2471i & 0.0283 + 0.3719i & -0.0641 - 0.5212i \\ 0.5363 + 0.0211i & -0.4133 - 0.3698i & 0.4294 - 0.2661i & -0.3389 + 0.1852i \end{bmatrix}, \\
B_5 &= \begin{bmatrix} 0.1610 - 0.0104i & -0.1781 - 0.0118i & -0.3241 + 0.4502i & 0.5883 - 0.5370i \\ -0.1163 - 0.7159i & -0.4619 + 0.0427i & -0.3587 + 0.1576i & -0.2228 + 0.2359i \\ 0.4255 + 0.4822i & -0.1864 + 0.4173i & -0.2767 + 0.3474i & -0.3281 + 0.2696i \\ -0.0127 + 0.1852i & -0.3440 - 0.6525i & 0.3753 + 0.4480i & 0.0114 + 0.2825i \end{bmatrix}, \\
B_6 &= \begin{bmatrix} 0.5947 + 0.0898i & 0.0777 - 0.0544i & 0.3469 - 0.1757i & -0.6284 + 0.2883i \\ 0.3998 + 0.2454i & 0.1581 - 0.4791i & 0.2068 + 0.3971i & 0.5486 + 0.1550i \\ 0.6131 - 0.1431i & 0.2198 + 0.2581i & -0.6078 + 0.0618i & 0.0583 - 0.3347i \\ 0.1429 + 0.0378i & -0.6927 + 0.3764i & -0.0616 + 0.5208i & -0.0268 + 0.2844i \end{bmatrix}, \\
B_7 &= \begin{bmatrix} 0.5078 + 0.1352i & 0.0293 - 0.6221i & -0.3714 - 0.0843i & 0.1771 - 0.3996i \\ -0.0888 + 0.0556i & -0.0162 + 0.1379i & -0.8660 - 0.0570i & -0.1509 + 0.4401i \\ 0.4760 - 0.1213i & -0.0101 - 0.3142i & 0.2954 - 0.0688i & -0.1032 + 0.7465i \\ -0.6725 - 0.1394i & 0.1043 - 0.6951i & 0.0136 + 0.0976i & -0.1192 + 0.1021i \end{bmatrix}, \\
B_8 &= \begin{bmatrix} 0.2398 - 0.2323i & 0.2102 + 0.2519i & -0.0454 + 0.7652i & 0.4128 + 0.1513i \\ -0.4485 + 0.0876i & 0.4478 + 0.3515i & -0.2440 - 0.3668i & 0.5225 - 0.0054i \\ 0.2040 - 0.3878i & 0.6345 + 0.2146i & -0.1047 - 0.1151i & -0.5780 - 0.0302i \\ 0.5155 - 0.4710i & -0.0559 - 0.3413i & -0.1917 - 0.3967i & 0.4298 - 0.1182i \end{bmatrix}, \\
B_9 &= \begin{bmatrix} -0.0808 + 0.1747i & 0.6974 + 0.0254i & -0.3297 + 0.1999i & -0.4411 + 0.3643i \\ -0.2270 - 0.1530i & 0.3884 + 0.1536i & 0.2603 + 0.6090i & 0.2922 - 0.4760i \\ -0.5193 + 0.7098i & -0.3679 + 0.1068i & -0.0637 + 0.2636i & 0.0287 + 0.0737i \\ 0.0413 - 0.3359i & -0.3492 + 0.2643i & 0.3891 + 0.4333i & -0.4673 + 0.3690i \end{bmatrix}, \\
B_{10} &= \begin{bmatrix} 0.6160 + 0.1969i & -0.4480 - 0.4219i & 0.1428 - 0.3612i & -0.2164 - 0.0733i \\ 0.2687 + 0.0204i & 0.6471 + 0.3236i & -0.0287 - 0.6065i & -0.1607 - 0.0976i \\ 0.0571 - 0.4298i & 0.0075 - 0.1040i & 0.0461 + 0.0462i & 0.2063 - 0.8685i \\ 0.5006 - 0.2658i & 0.2913 + 0.0464i & 0.3550 + 0.5917i & -0.3108 + 0.1379i \end{bmatrix}, \\
B_{11} &= \begin{bmatrix} 0.7748 - 0.0926i & -0.1322 - 0.1752i & 0.0966 + 0.3047i & 0.2966 + 0.3909i \\ -0.2424 - 0.3439i & 0.3463 - 0.1798i & 0.0991 + 0.7302i & -0.3426 + 0.1017i \\ 0.2561 - 0.1187i & -0.3777 - 0.5443i & 0.0569 - 0.1325i & -0.5445 - 0.4052i \\ 0.2411 - 0.2762i & 0.6003 + 0.0157i & 0.2016 - 0.5416i & -0.3096 + 0.2740i \end{bmatrix},
\end{aligned}$$

Table 1: Continuous Diagonal Code  $\Lambda = \text{diag}(\exp(\frac{2\pi i}{L}u))$ 

$M$	$R$	$L$	$u$
2	1	4	[1 1.6741]
3	1	8	[1 1.9537 2.9759]
4	1	16	[1 2.9976 5.0063 6.9979]
5	1	32	[1 2.8963 7.9168 12.3396 14.1375]
6	1	64	[1 3.9663 5.8291 17.8483 24.6302 26.5638]
7	1	128	[1 3.9607 21.9899 31.5332 47.3852 54.2734 60.2040]
2	2	16	[1 5.9911]
3	2	64	[1 6.8881 26.5877]
4	2	256	[1 7.9761 68.6816 106.6000]
5	2	1024	[1 61.0483 100.6309 129.7491 356.4678]
6	2	4096	[1 11.8659 404.3640 592.2112 1328.7582 1489.9040]
7	2	16384	[1 300.8485 4019.3073 5142.8482 6816.8842 8098.6177 8109.4273]

$$\begin{aligned}
 B_{12} &= \begin{bmatrix} 0.1280 + 0.2465i & -0.6345 - 0.1417i & -0.0766 - 0.4249i & 0.0486 - 0.5581i \\ 0.0816 - 0.6158i & 0.2517 - 0.0380i & 0.0347 + 0.2233i & 0.1347 - 0.6929i \\ 0.2862 - 0.1410i & -0.3924 + 0.2196i & 0.7920 + 0.2295i & -0.0725 + 0.1040i \\ -0.2304 - 0.6181i & -0.5369 - 0.1485i & -0.2868 + 0.0181i & 0.1966 + 0.3650i \end{bmatrix}, \\
 B_{13} &= \begin{bmatrix} 0.0256 - 0.1183i & 0.1067 + 0.2697i & 0.1343 - 0.1014i & 0.9325 - 0.0583i \\ -0.5312 - 0.2629i & -0.6552 - 0.4068i & -0.0562 - 0.1474i & 0.1628 - 0.0512i \\ 0.5708 - 0.4629i & -0.4947 + 0.2483i & 0.1888 + 0.1781i & -0.1143 - 0.2703i \\ 0.0258 + 0.3055i & -0.0856 - 0.0859i & 0.8342 - 0.4284i & -0.0972 - 0.0495i \end{bmatrix}, \\
 B_{14} &= \begin{bmatrix} 0.3080 - 0.2933i & 0.4685 - 0.1297i & 0.2442 - 0.6850i & 0.1051 - 0.2072i \\ 0.0902 - 0.4068i & 0.0109 + 0.4103i & 0.0850 + 0.1134i & -0.7756 - 0.1903i \\ -0.1509 + 0.2560i & 0.7121 - 0.2210i & -0.0907 + 0.4222i & -0.1097 - 0.3966i \\ 0.7444 + 0.0551i & 0.1963 + 0.0281i & -0.4936 + 0.1449i & -0.0483 + 0.3696i \end{bmatrix}, \\
 B_{15} &= \begin{bmatrix} 0.2287 + 0.2115i & 0.4099 - 0.4573i & -0.6287 + 0.2885i & -0.2128 - 0.0457i \\ -0.3966 - 0.7004i & -0.0114 - 0.2050i & -0.4089 - 0.1241i & 0.2918 + 0.2056i \\ -0.1676 + 0.2912i & 0.4021 - 0.4722i & 0.4131 - 0.1470i & 0.3944 + 0.3933i \\ -0.3760 - 0.0294i & -0.1257 - 0.4246i & 0.1711 - 0.3427i & -0.7080 - 0.1175i \end{bmatrix}.
 \end{aligned}$$

## F STRUCTURE OF ORTHOGONAL DESIGNS

In our simulation of orthogonal designs, the frame length is chosen as  $T \geq M$  and the transmitted signals are determined as

$$S_0 = \sqrt{\frac{T}{M}} \begin{bmatrix} I_{M \times M} \\ 0_{(T-M) \times M} \end{bmatrix}, \quad S_\tau = V_\tau S_{\tau-1}, \quad \tau = 1, 2, \dots$$

where  $S_\tau$  is a  $T \times M$  matrix,  $V_\tau = \mathcal{G}(z_1, \dots, z_K)$  is defined by a  $T \times T$  orthogonal design  $\mathcal{G}$  such that  $z_1, \dots, z_K$  are mapped from PSK constellations  $\mathcal{A}_1, \dots, \mathcal{A}_K$ . The choice of  $T$  depends on the number transmit antennas. For  $M = 2$  we choose  $T = 2$  and use the  $2 \times 2$  orthogonal design in [3]. For  $M = 3$  and 4, we choose  $T = 4$  and use the  $4 \times 4$  orthogonal design in [47]. For  $M = 5, 6, 7, 8$ , we chose  $T = 8$  and use the  $8 \times 8$  orthogonal design in [47]. It should be noted that such concatenation between the complex square orthogonal designs and the differential unitary space-time modulation scheme has been proposed in [13] and [28]. For the spectral efficiency to be an integer  $R$ , we use the following PSK constellations

$$\mathcal{A}_k = \left\{ \frac{1}{\sqrt{K}} \exp \left( j \frac{2\pi r}{\lceil \frac{TR}{K} \rceil} \right) \mid r = 0, 1, \dots, 2^{\lceil \frac{TR}{K} \rceil} - 1 \right\}$$

for  $1 \leq k \leq (TR \bmod(K))$ ;

$$\mathcal{A}_k = \left\{ \frac{1}{\sqrt{K}} \exp \left( j \frac{2\pi r}{\lfloor \frac{TR}{K} \rfloor} \right) \mid r = 0, 1, \dots, 2^{\lfloor \frac{TR}{K} \rfloor} - 1 \right\}$$

for  $(TR \bmod(K)) < k \leq K$ . For the  $\tau$ -th time frame, bits of length  $TR$  are mapped into  $z_k \in \mathcal{A}_k$ ,  $k = 1, \dots, K$  by Gray codes. The decoding problem is to solve the minimization problem

$$\arg \min_{z_k \in \mathcal{A}_k, k=1, \dots, K} \|X_\tau - \mathcal{G}(z_1, \dots, z_K) X_{\tau-1}\|_F^2. \quad (68)$$

As demonstrated in [45], by exploiting the special structure of the orthogonal design, the data symbols  $z_k \in \mathcal{A}_k$ ,  $k = 1, \dots, K$  can be decoupled and decoded individually from (68).

## Acknowledgment

The authors wish to thank the anonymous reviewers for their valuable comments and suggestions.

## References

- [1] D. Agrawal, T. J. Richardson, and R. L. Urbanke, "Multiple-antenna signal constellation for fading channels," *IEEE Trans. Inform. Theory*, vol. 47, pp. 2618-2626, Sept. 2001.
- [2] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Trans. Inform. Theory*, vol. 48, pp. 2201-2214, Aug. 2002.
- [3] S. M. Alamouti, "A simple transmitter diversity scheme for wireless communications," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 1451-1458, Oct. 1998.
- [4] M. J. Borran, A. Sabharwal, and B. Aazhang, "On design criteria and construction of noncoherent space-time constellations," *IEEE Trans. Inform. Theory*, vol. 49, pp. 2332-2351, Oct. 2003.
- [5] K. L. Clarkson, W. Sweldens, and A. Zheng, "Fast multiple antenna differential decoding," *IEEE Trans. Commun.*, vol. 49, pp. 253-261, Feb. 2001.
- [6] M. O. Damen, A. Chkeif, and J.-C. Belfiore, "Lattice code decoder for space-time codes," *IEEE Commun. Lett.*, pp. 161-163, May 2000.
- [7] M. O. Damen, K. Abed-Meraim, and J.-C. Belfiore, "Diagonal algebraic space-time block codes," *IEEE Trans. Inform. Theory*, vol. 48, pp. 628-636, Mar. 2002.
- [8] M. O. Damen, H. El Gamal, and G. Caire, "On maximum-likelihood detection and the search for the closest lattice point," *IEEE Trans. Inform. Theory*, vol. 49, pp. 2389- 2402, Oct. 2003.
- [9] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," *Math. Comput.*, vol. 44, pp. 463-471, Apr. 1985.
- [10] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Personal Commun.*, vol. 6, pp. 311-335, Mar. 1998.
- [11] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs. Tech. J.*, vol. 1, no. 2, pp. 41-59, 1996.

- [12] H. El Gamal and A. R. Hammons Jr, "On the design and performance of algebraic space-time codes for BPSK and QPSK modulation," *IEEE Trans. Commun.*, vol. 50, pp. 907-913, June 2002.
- [13] G. Ganesan and P. Stoica, "Differential modulation using space-time block codes," *IEEE Signal Processing Letters*, vol. 9, no. 2, pp. 57-60, Feb. 2002.
- [14] J.-C. Guey, M. P. Fitz, M. R. Bell, and W.-Y. Kuo, "Signal design for transmitter diversity wireless communication systems over Rayleigh fading channels," *IEEE Trans. Commun.*, vol. 47, pp. 527-537, Apr. 1999.
- [15] D. Gesbert, M. Shafi, D. Shiu, P. Smith, and A. Naguib, "From theory to practice: an overview of MIMO space-time coded wireless systems," *IEEE J. Select. Areas Commun.*, vol. 21, pp. 281-301, Apr. 2003.
- [16] B. Hassibi and B. Hochwald, "Caley differential unitary space-time Codes," *IEEE Transactions on Information Theory*, vol. 48, pp. 1485-1503, June 2002.
- [17] B. Hassibi and B. Hochwald, "High-rate codes that are linear in space and time," *IEEE Trans. Inform. Theory*, vol. 48, pp. 1804-1824, July 2002.
- [18] B. Hassibi, T. Marzetta, and B. Hochwald, "Structured unitary space-time autocoding constellations," *IEEE Trans. Inform. Theory*, vol. 48, pp. 942-950, Apr. 2002.
- [19] B. Hochwald, T. Marzetta, T. Richardson, W. Sweldens, and R. Urbanke, "Systematic design of unitary space-time constellations," *IEEE Trans. Inform. Theory*, vol. 46, pp. 1962-1973, Sept. 2000.
- [20] B. Hochwald and W. Sweldens, "Differential unitary space time modulation," *IEEE Trans. Commun.*, vol. 48, pp. 2041-2052, Dec. 2000.
- [21] B. M. Hochwald and T. L. Marzetta, "Unitary space-time modulation for multiple-antenna communication in Rayleigh flat-fading," *IEEE Trans. Inform. Theory*, vol. 46, pp. 543-564, Mar. 2000.
- [22] B. Hughes, "Differential space-time modulation," *IEEE Trans. Inform. Theory*, vol. 46, pp. 2567-2578, Nov. 2000.

- [23] B. Hughes, "Optimal space-time constellations from groups," *IEEE Trans. Inform. Theory*, vol. 49, pp. 401-410, Feb. 2003.
- [24] C.-S. Hwang, S. H. Nam, J. Chung, and V. Tarokh, "Differential space time block codes using nonconstant modulus constellations," *IEEE Trans. on Signal Processing*, vol. 51, no. 11, pp. 2955-2964, Nov. 2003.
- [25] H. Jafarkani and V. Tarokh, "Multiple transmit antenna differential detection from generalized orthogonal design," *IEEE Trans. Inform. Theory*, vol. 47, pp. 2626-2631, Sept. 2001.
- [26] A. K. Lenstra, H. W. Lenstra, and L. Lovász, "Factoring polynomials with rational coefficients," *Math. Ann.*, vol. 261, pp. 515-534, 1982.
- [27] X. B. Liang and X. G. Xia, "Unitary signal constellations for differential space-time modulation with two transmit antennas: Parametric codes, optimal designs and bounds," *IEEE Trans. Inform. Theory*, vol. 48, pp. 2291-2322, Aug. 2002.
- [28] X. B. Liang and X. G. Xia, "Fast differential unitary space-time demodulation via square orthogonal designs," to appear in *IEEE Transaction on Wireless Communications*.
- [29] X. B. Liang, "Orthogonal designs with maximal rates," *IEEE Trans. Inform. Theory*, vol. 49, pp. 2468-2503, Oct. 2003.
- [30] Z. Liu, G. Giannakis, and B. Hughes, "Double differential space-time block coding for time-selective fading channels," *Proc. IEEE WCNC*, 2000.
- [31] T. L. Marzetta and B. M. Hochwald, "Capacity of a mobile multiple-antenna communication link in Rayleigh flat fading," *IEEE Trans. Inform. Theory*, vol. 45, pp. 139-157, Jan. 1999.
- [32] M. L. McCloud, M. Brehler, and M. K. Varanasi, "Signal design and convolutional coding for noncoherent space-time communication on the block-Rayleigh-fading channel," *IEEE Trans. Inform. Theory*, vol. 48, pp. 1186-1194, May 2002.
- [33] F. D. Murnaghan, *The Unitary and Rotation Groups, vol. III of Lectures on Applied Mathematics*, Washington, DC: Spartan, 1962.
- [34] A. Narula, M. D. Trott, and G. W. Wornell, "Performance limits of coded diversity methods for transmitter antenna arrays," *IEEE Trans. Inform. Theory*, vol. 45, pp. 2418-2433, Nov. 1999.



- [35] A. F. Naguib, N. Seshadri, and A. R. Calderbank, "Space-time coding and signal processing for high data wireless communications," *IEEE Signal Processing Mag.*, vol. 17, no. 3, pp. 76-92, 2000.
- [36] G. G. Raleigh and J. M. Cioffi, "Spatial-temporal coding for wireless communication," *IEEE Trans. Commun.*, vol. 46, pp. 357-366, Mar. 1998.
- [37] C. P. Schnorr and M. Euchner, "Lattice basis reduction: Improved practical algorithms and solving subset sum problems," *Math. Programming*, vol. 66, pp. 181-191, 1994.
- [38] N. Seshadri and J. H. Winters, "Two signalling schemes for improving the error performance of frequency-division-duplex (FDD) transmission systems using transmitter antenna diversity," *Int. J. Wireless Inform. Networks*, vol. 1, no. 1, pp. 49-59, 1994.
- [39] A. Shokrollahi, B. Hassibi, B. Hochwald, and W. Sweldens, "Representation theory for high-rate multiple-antenna code design," *IEEE Trans. Inform. Theory*, vol. 47, pp. 2335-2367, Sept. 2001.
- [40] M. Tao and R. S. Cheng, "Trellis-coded differential unitary space-time modulation over flat fading channels," *IEEE Trans. Commun.*, vol. 51, pp. 587-596, Apr. 2003.
- [41] M. Tao and R. S. Cheng, "Differential space-time block codes," *IEEE GLOBECOM*, vol. 2, pp. 1098-1102, 2001.
- [42] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: Performance criterion and code construction," *IEEE Trans. Inform. Theory*, vol. 44, pp. 744-765, Mar. 1998.
- [43] V. Tarokh, H. Jafarkani, and A. R. Calderbank, "Space-time block codes from orthogonal designs," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1456-1467, July 1999.
- [44] V. Tarokh and H. Jafarkhani, "A differential detection scheme for transmit diversity," *J. Select. Areas Commun.*, pp. 1169-1174, July 2000.
- [45] V. Tarokh, H. Jafarkani, and A. R. Calderbank, "Space-time block coding for wireless communications: performance results," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 451-460, Mar. 1999.

- [46] E. Teletar, "Capacity of multi-antenna Gaussian channels," *European Transaction on Telecommunications*, vol. 6, Nov. - Dec. 1999, pp. 585-595.
- [47] O. Tirkkonen and A. Hottinen, "Square-matrix embeddable space-time block codes for complex signals constellations," *IEEE Trans. Inform. Theory*, vol. 48, pp. 384-395, Feb. 2002.
- [48] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channel," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1639-1642, July 1999.
- [49] D. Warrier and U. Madhow, "Spectrally efficient noncoherent communications," *IEEE Trans. Inform. Theory*, vol. 48, pp. 651-668, Mar. 2001.
- [50] J. H. Winters, "Switched diversity with feedback for DPSK mobile radio systems," *IEEE Trans. Veh. Technol.*, vol. 32, pp. 134-150, 1983.
- [51] J. H. Winters, "Diversity gain of transmit diversity in wireless systems with Rayleigh fading," in *Proc. IEEE Int. Communications Conf.*, vol. 2, pp. 1121-1125, 1994.
- [52] A. Wittneben, "Base station modulation diversity for digital simulcast," *Proc. IEEE Vehicular Technology Conf.*, pp. 505-511, May 1993.
- [53] A. Wittneben, "A new bandwidth efficient transmit antenna modulation diversity scheme for linear digital modulation," *Proc. ICC93*, pp. 1630-1634.
- [54] X. G. Xia, "Differentially en/decoded orthogonal space-time block codes with APSK signals," *IEEE Commun. Lett.*, vol. 6, pp. 150-152, Apr. 2002.
- [55] L. Zheng and D. N. C. Tse, "Communication on the Grassmann manifold: A geometric approach to the noncoherent multiple-antenna channel," *IEEE Trans. Inform. Theory*, vol. 48, pp. 359-383, Feb. 2002.
- [56] D. Zwillinger, *CRC Standard Mathematical Tables and Formulae*, pp. 691-692, CRC Press, 1996.